Multi-fidelity methods for fluid-structure interaction and uncertainty quantification

Multi-fidelity methods for fluid-structure interaction and uncertainty quantification

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus Prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
woensdag 30 september 2015 om 12.30 uur

door

Thomas Peter Scholcz

ingenieur Luchtvaart- en Ruimtevaarttechniek
geboren te Nieuwegein.

This dissertation has been approved by the
promotor: Prof. dr. drs. H. Bijl
copromotor: Dr. ir. A.H. van Zuijlen

Composition of the doctoral committee:

Rector Magnificus	chairman
Prof. dr. ir. drs. H. Bijl	Delft University of Technology
Dr. ir. A.H. van Zuijlen	Delft University of Technology

Independent members:

Prof. dr. ir. C. Vuik	Delft University of Technology
Prof. dr. ir. B. Koren	Eindhoven University of Technology
Prof. dr. ir. J. Vierendeels	Ghent University
Prof. dr. ir. C. Lacor	Free University of Brussels
Prof. dr. S. Hickel	Delft University of Technology

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7 / 2007-2013) under a grant agreement number 233665. Future Fast Aeroelastic Simulation Technologies (FFAST) is a collaborative research project aimed at developing, implementing and assessing a range of numerical simulation technologies to accelerate future aircraft design. The partners in FFAST are: University of Bristol, INRIA, CSIR, TU Delft, DLR, IRIAS, University of Liverpool, Politecnico di Milano, NUMECA, Optimad Engineering, Airbus, EADS-MS, IITP and UCT.

Keywords: Multi-fidelity, uncertainty quantification, fluid-structure interaction, space-mapping, kriging, adjoints, non-intrusive

Printed by: Sieca Repro Delft

Front & Back: Richard Steiner

ISBN 978-94-6259-848-5

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

Perhaps if one flew high enough ... one would be able to see

J.M. Coetzee

Contents

Summary

The cost and turnaround time of the load calculation cycle in the design process of aircraft can be reduced by developing new numerical simulation technologies aimed at efficient prediction of steady and unsteady force coefficients in the flight envelope. In order to capture the most extreme stress levels due to gust and manoeuvres, efficient time-accurate analysis of fluid-structure interaction is desired. In order to incorporate uncertainties in flight conditions, manoeuvres, shape and material properties, efficient uncertainty quantification is desired.

This thesis aims at developing efficient multi-fidelity algorithms for fluid-structure interaction and uncertainty quantification. Space-mapping is a multi-fidelity technique that can be applied in order to accelerate strongly coupled partitioned fluid-structure interaction. Gradient-Enhanced Kriging is a response surface technique that can be used for uncertainty quantification.

Aggressive Space-Mapping is applied to three academic fluid-structure interaction problems of increasing complexity. For most test cases considered the estimated and observed speedup with respect to the Quasi-Newton algorithm is larger than 1. The influence of the timestep size on the speedup is large compared to other parameters. In case of compressible flows the speedup can rise to 1.5 for large time-steps. In case of incompressible flows higher speedups can be expected due to strong coupling of the fluid-structure interaction.

Gradient-Enhanced Kriging (GEK) is applied to two academic uncertainty quantification problems of increasing complexity. The observed speedup of GEK with respect to Kriging increases with the number of dimensions of the design-space. When the gradients are computed with the adjoint method the theoretical speedup is $S = \frac{1}{2}(1 + d)$, where d is the number of dimensions. The observed speedup and theoretical speedup are only close if the derivative noise, variation of the sampling plan and the variation of the target accuracies are taken into account.

Samenvatting

De kosten en omlooptijd van het berekenen van belastingen binnen het ontwerpproces van vliegtuigen kan verminderd worden door de ontwikkeling van nieuwe numerieke simulatie technologieën toegespitst op het efficiënt voorspellen van stationaire en tijdafhankelijke kracht coëfficiënten binnen de uiterste operationele begrenzingen. Het ondervangen van de meest extreme spanningswaarden veroorzaakt door windvlagen en manoeuvreerbewegingen vraagt om tijdsnauwkeurige analyse van de interactie tussen vloeistof/gas en structuur. Om onderzekerheden in vliegomstandigheden, manoeuvreerbewegingen, vorm en materiaaleigenschappen te kunnen ondervangen is efficiënte onzekerheidsanalyse nodig.

Deze dissertatie richt zich op de ontwikkeling van efficiënte algoritmen ten behoeve van vloeistof/gas - structuur interactie en onzekerheidsanalyse, gebruikmakend van modellen met een verschillende betrouwbaarheid. Space-mapping is een techniek die toegepast kan worden om sterk gekoppelde en gepartitioneerde vloeistof/gas - structuur interactie te versnellen. Gradient-Enhanced Kriging is een responsieoppervlak techniek die toegepast kan worden om onzekerheid te kwantificeren.

Aggressive Space-Mapping is toegepast op drie academische vloeistof/gas - structuur interactie problemen van toenemende complexiteit. De geschatte en waargenomen versnelling ten opzichte van de Quasi-Newton techniek zijn groter dan 1 voor de meerderheid van de berekeningen. De invloed van de tijdstap op de versnelling is groot ten opzichte van de invloed van andere parameters. De versnelling kan toenemen tot 1.5 in het geval van compressibele stroming en grote tijdstappen. Grotere versnellingen kunnen worden verwacht voor incompressibele stromingen door de sterkere interactie tussen vloeistof en gas.

Gradient-Enhanced Kriging (GEK) is toegepast op twee academische onzekerheidsproblemen van toenemende complexiteit. De waargenomen versnelling van GEK ten opzichte van Kriging neemt toe met het aantal dimensies van de ontwerpruimte. De theoretische versnelling is gelijk aan $S = \frac{1}{2}(1 + d)$ wanneer de gradiënten worden uitgerekend middels het oplossen van de geadjugeerde vergelijkingen, met d het aantal dimensies van de ontwerpruimte. De waargenomen en de theoretische versnelling liggen alleen dicht bij elkaar wanneer rekening wordt gehouden met de numerieke ruis in de gradiënten, de variatie van het ontwerp-experiment en de variatie van de doelnauwkeurigheden.

1

Introduction

In this thesis, multi-fidelity algorithms are developed in order to reduce the computational cost of fluid-structure interaction simulations and Uncertainty Quantification (UQ). The research leading to the results in this thesis has received funding from the European Community's Seventh Framework Programme. Future Fast Aeroelastic Simulation Technologies (FFAST) is a collaborative research project aimed at developing, implementing and assessing a range of numerical simulation technologies to accelerate future aircraft design. A description of the FFAST project is found in section 1.1. An introduction to multi-fidelity acceleration of high fidelity fluid-structure simulations is given in section 1.2, followed by an introduction to meta-model based Uncertainty Quantification in section 1.3.

1.1. Future Fast Aeroelastic Simulation Technologies

Reducing the cost and turnaround time of the loads process within the design cycle of future aircraft will lead to significant improvements to product development and manufacture. Unsteady loads calculations play an important role within the loads process and have a large impact on the conceptual and detailed design, weight, aerodynamic performance, control characteristics etc. Especially loads cases due to gusts and manoeuvres are important since they determine the most extreme stress levels, fatigue damage and damage tolerance during the design cycle.

The actual flight conditions and manoeuvres during flight are highly uncertain. For this reason a large number of conditions need to be considered during the design cycle. Each flight condition/manoeuvre requires expensive numerical simulations in order to estimate the resulting stress levels in the aircraft structure. Each time the design of the aircraft is updated the process is repeated, resulting in a very expensive analysis. Nowadays, a loads calculation cycle requires more than 6 weeks.

1

The cost and turn around time of the load calculation cycle can be reduced by developing new efficient numerical simulation technologies. Improving the efficiency of numerical simulation technologies leads to [1, 2]:

- A decrease in wind tunnel testing requirements.
- A decreased risk of design modification later in the design process.
- A reduction of in-flight testing requirements.
- A faster certification process.
- An improvement of safety by enabling the design of systems to improve the response to extreme situations such as gust loads, wind shear and clear air turbulence.

The vision is illustrated in Figure 1.1. In this figure the representation fidelity is the fidelity of the final product represented in a scale from zero to one and the Virtual Product (VP) is a mathematical/numerical representation of the physical properties of a product. From Figure 1.1 it becomes clear that the product development is not

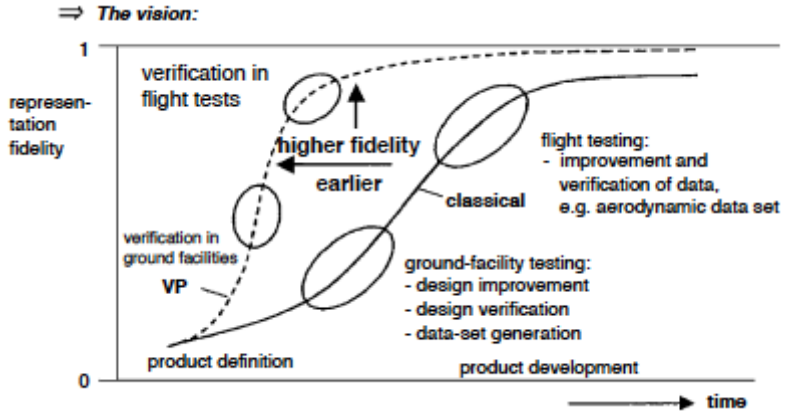


Figure 1.1: The vision: Using high fidelity solvers earlier in the design-cycle. Source: *Navier-Stokes solvers in European aircraft design*, Progress in Aerospace Sciences 38, (2002) 601-697.

only accelerated by taking high fidelity numerical simulations earlier into account, it also results in a *product with a higher representation fidelity* [2]. High fidelity analysis is furthermore required due to

- More demanding future performance targets, pushing the design towards the envelope boundaries.
- The lack of engineering experience with novel configurations: In this case experience cannot be used to identify the important areas in the flight envelope such that the number of critical loads cases are reduced.

Three major areas of research to reduce the total analysis costs have been identified in the FFAST project, [1]:

- Identification of critical loads cases: The number of conditions/manoeuvres can be reduced by identifying the most important loads cases, see [3–6]. This work focuses on prediction of the worst-case gust loads that a particular aircraft might encounter during flight.
- Multi-fidelity acceleration of high fidelity models: In this case low fidelity models are exploited in convergence acceleration methods. This results in so called hybrid methods, see [7, 8].
- The construction of meta-models from high fidelity models. The meta-models can then be used to do the high fidelity loads analysis at a lower cost. Techniques used to build meta-models are for example Neural Networks [9], Radial Basis Functions or Kriging [3, 5], Multivariate Adaptive Regression Splines [5] and Polynomial Chaos [10, 11].

The main Quantities of Interest (QoI) are the distributed forces around the aircraft as well as integral quantities such as the lift, drag or moment coefficients, see figure 1.2. Efficient prediction of the steady and unsteady force coefficients in the flight envelope is the main objective.

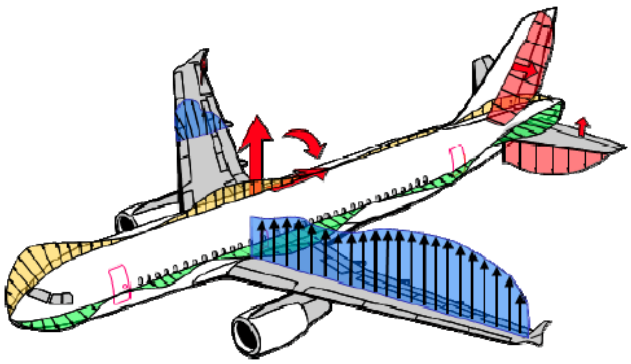


Figure 1.2: Distributed loads and integral quantities on an aircraft. Source: Airbus

Uncertainties in flight conditions, manoeuvres, shape and material properties lead furthermore to the development of methods that quantify these uncertainties. Taking uncertainties into account increases the representation fidelity of the final product even more. Of interest are statistical moments of the QoI and the probability of occurrence of undesired phenomena. Uncertainty and reliability analysis of fluid-structure stability boundaries have been investigated in [12–15]. The uncertainty propagation methods in these contributions are mainly Monte Carlo, perturbation and interval analysis. Monte Carlo methods are generic and accurate but require too many samples to reach acceptable target accuracies. Perturbation

methods, on the other hand, are cheap but only give acceptable accuracy when the problem is sufficiently linear. Especially at the envelope boundaries where the Mach regime is mainly transonic, non-linearities cannot be ignored and should be taken into account.

Meta-models / response surface methods are designed to capture the most important non-linearities of the high fidelity model. The use of meta-models in uncertainty quantification and reliability analysis has therefore become an active field of research [16–19]. These methods are only successful if the constructed meta-models are sufficiently accurate in the region of interest. The use of a meta-model implies a loss in accuracy since the accuracy of the meta-model can never match the accuracy of the high fidelity model from which it was constructed. Two approaches exist to cope with the loss of accuracy:

1. Meta-model error estimation: Examples are the delta method [20], cross validation [21] and bootstrapping [22]. Estimation of the meta-model error allows to construct meta-models with the desired target accuracy.

2. The use of hybrid methods: The meta-model is used to accelerate the analysis on the high fidelity model such that there is no loss of accuracy. Examples are meta-model based importance sampling [23] or meta-model enhanced Monte Carlo sampling [24].

This dissertation is divided into two parts. The first part concentrates on multi-fidelity acceleration of high fidelity fluid-structure interaction simulations. An introduction on fluid-structure interaction is given followed by the discussion of several coupling techniques. Finally, a new coupling technique is introduced based on space-mapping which is a mathematical technique originating from the field of multi-fidelity optimization. The second part contains work on meta-model based uncertainty quantification. Here, the focus is on the efficient construction of meta-models. An introduction on meta-model based uncertainty quantification is given starting with the classical perturbation method followed by the Kriging method and the crude Monte Carlo method.

1.2. Multi-fidelity acceleration of high fidelity fluid-structure interaction simulations

Fluid-Structure Interaction (FSI) is the mutual interaction between a fluid and a deformable structure. When a fluid is in interaction with a structure it exerts forces on it such that the structure deforms. The deformation of the structure will in turn affect the fluid flow such that a coupled problem results.

Fluid-Structure Interactions play a central role in aerospace engineering and many other fields like civil, mechanical and biomedical engineering [25, 26]. Unstable interactions like wing flutter and buffeting can cause structural failure and prediction of their occurrence is of primary importance in the design of aircraft [12]. Asymptotic stability is a necessary but insufficient condition to guarantee structural integrity. It has been shown that transient growth, induced by sources of external

excitation such as gust loads, can lead to structural failure despite the fact that the system is asymptotically stable [13, 27]. Unsteady high fidelity simulations can overcome the shortcomings of steady analysis but it is computationally much more expensive. A time-accurate CFD analysis with a dual-time stepping scheme is roughly a hundred times more expensive as a single steady calculation of the same accuracy [28]. The "brute force" analysis of 50 flight points in the flight envelope for 100 mass cases, 10 control surface configurations, 50 manoeuvres, and 4 control laws results in 10,000,000 unsteady high fidelity simulations to perform a single load calculation cycle. The total analysis cost of a "brute force" load calculation cycle is therefore estimated to be roughly 1000,000,000 times more expensive as a single steady CFD analysis. Reducing the computational effort of unsteady high fidelity simulations - often involving fluid-structure interaction - is therefore of utmost importance.

1.2.1. Simulation of high fidelity fluid-structure interaction

The replacement of low fidelity simulations with more accurate FSI simulations is attractive because it reduces the number of design cycles, the development risk, the number of flight tests, the cost and time to market and the risk of design modifications in the later design phases [2]. However, the computational effort associated with high fidelity FSI models currently precludes their direct use in industry. Acceleration of time-accurate high fidelity aeroelastic simulation algorithms has therefore become an active area of research. The black-box approach is especially attractive due to the minimal intrusiveness and modularity of the resulting algorithms. Using an implicit time integration scheme the following residual equation needs to be solved at each time step in the simulation

$$\mathcal{R}(\mathbf{x}) \equiv \mathcal{H}(\mathbf{x}) - \mathbf{x} = 0,$$

where $\mathcal{H} = \mathcal{S} \circ \mathcal{F}$ contains the structure and fluid operator, which can be seen as black-boxes since they only have an input/output structure. Solving the residual equation implies that the kinematic and dynamic interface conditions on the fluid-stucture interface are satisfied. Examples of black-box coupling algorithms that aim to solve the residual equation effciently are multi-level approaches [29–32], multi-solver approaches [33], Interface-GMRES(R) [34, 35], Aitken's method and vector extrapolation [36, 37] and the Quasi-Newton Inverse Least Squares (QN-ILS) method [33, 38–40].

The QN-ILS method has become a popular method due to its combination of efficiency and simplicity, see [33, 38, 40] and its thorough theoretical basis, see [39]. In [40] it was found that the QN-ILS method outperforms Aitken's method and the Newton-Krylov method from [34] when applied to a (nonlinear) strongly coupled FSI problem. In [39] it was found that the QN-LS method is only slightly slower than GMRes when applied to obtain the solution of several linear systems of equations and in [41] it is shown that the QN-ILS method can be modified to become analytically equivalent to GMRES. A general comparison of various partitioned iterative solution methods for FSI is found in [42, 43].

1

1.2.2. Aim and motivation

Strong coupling algorithms are designed to efficiently solve the coupled problem that results from an implicit time integration scheme applied to the semi-discrete system of equations describing the fluid and solid dynamics, the so called *partitioned* approach. The partitioned approach allows software modularity and reuse of existing field solvers and is therefore more promising in an industrial environment than the *monolithic* approach, which aims at solving the fluid and solid systems simultaneously. In the transonic regime, the flow interacts strongly with the structure since the flow is highly nonlinear and very sensitive to structural motions [44]. Especially for large time steps in the transonic regime, strong coupling procedures are necessary in order to avoid excessive phase-lag errors [45]. Strong coupling algorithms are more expensive but unavoidable since loosely coupled algorithms yield unacceptable accuracy in this regime.

- The fact that loosely coupled algorithms are inaccurate due to the dominating partitioning error for large time steps in the transonic regime motivates the development of more efficient strong partitioned coupling algorithms.

Furthermore, we focus on problems with the following property:

- The high fidelity fluid solver consumes much more CPU time than the high fidelity structure solver.

This is typical in aerospace applications [46]. Reducing the number of fluid solves per time step while maintaining accuracy is therefore the main objective in the development of new strong coupling algorithms. We investigate the use of low fidelity models to speed up partitioned coupling simulations applied to high fidelity models, the so called *multi-fidelity* approach. Without loss of generality we assume that two solvers are available: a cheap low fidelity fluid solver and an expensive high fidelity fluid solver, see Figure 1.3.

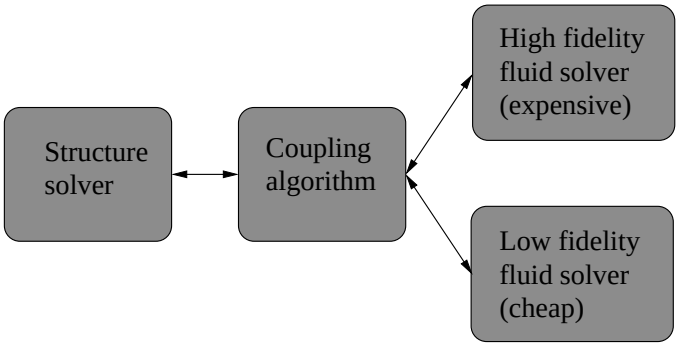


Figure 1.3: Schematic of a multi-fidelity coupling algorithm.

In principle we could also introduce a cheap low-fidelity structure solver in the problem. Since we are mainly focused on reducing the number of high fidelity fluid

solves this option is not considered here. The research objective is to increase the efficiency of high fidelity fluid-structure interaction simulations by means of multi-fidelity coupling. In order to achieve this goal algorithms need to be developed that take into account information from the low fidelity model in the high fidelity model computation. In addition to the main objective several constraints are identified. In summary the research objective and additional constraints are:

Main objective: Improve the efficiency of high fidelity fluid-structure interaction simulations using multi-fidelity coupling.

Constraints:

1. **Software modularity**: The solution strategy should be able to couple existing field solvers for both the fluid and the structure. This requirement ensures that the Computational Fluid Dynamics and Computational Structural Mechanics communities can develop their own codes using their own expertise. A modular approach treats the field solvers as modules that are coupled by exchanging information through interfaces. The algorithms are still useful when better mathematical models and methods are developed.

2. **Minimal intrusiveness**: Algorithms need to be developed that are able to couple existing fluid and structure codes in a minimal intrusive way. The resulting coupling algorithms should have a high numerical efficiency and stability without having to adapt the existing codes too much.

3. **Robustness**: In addition to the numerical efficiency and stability requirements it is also important to develop a solution procedure that can be used for a wide variety of physical parameters. Robustness is the sensitivity of the convergence of a method to variation in problem parameters. This requirement is especially important when different configurations of aerospace vehicles need to be tested at a wide range of flight conditions.

4. **Generality**: The solution procedure should be able to couple a wide variety of low fidelity models to high fidelity fine models. Software modularity is therefore also required with respect to low fidelity models used to achieve the acceleration in the solution procedure of the high fidelity model. The resulting algorithm is generic and can benefit from the efforts invested by other parties in constructing a suitable low fidelity model. The low fidelity model could be a coarse mesh discretization of the original problem, a simplified mathematical model (e.g. a panel method) but also a low fidelity model that originates from a suitable Reduced Order Model (ROM) technique.

These four constraints are tightly inter-related, e.g. generality implies software modularity and minimal intrusiveness can only be achieved if the constraints of software modularity are satisfied. Robustness is inter-related with the main objective: if the efficiency increases it means that the ratio of accuracy over computational resources has changed which may affect the robustness of the algorithm.

It is therefore of crucial importance to find a balance between the main objective and the constraints that need to be satisfied. This is the art of designing a new successful algorithm.

1.2.3. Quasi-Newton methods

An important class of coupling methods that satisfy the constraints are the class of Quasi-Newton (QN) methods. To satisfy kinematic and dynamic interface conditions on the fluid-structure interaction interface the residual equation needs to be solved. Newton's method can be used to obtain

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\frac{\partial \mathcal{R}}{\partial \mathbf{x}}\right)^{-1} \mathcal{R}(\mathbf{x}^k).$$

The Jacobian of the residual in this expression is however unknown due to the black-box nature of the solvers. Quasi-Newton methods use different approximations to the Jacobian of the residual. Quasi-Newton updates are then used to solve the residual equation iteratively. As an alternative to the residual formulation, the fixed-point formulation can be used.

$$\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^k) \tag{1.1}$$

Fixed-point iterations are equivalent to quasi-Newton iterations if the Jacobian of the residual is approximated by the negative identity matrix. Quasi-Newton methods can therefore be seen as fixed-point acceleration schemes. On the other hand, fixed-point acceleration schemes like Aitken's adaptive underrelaxation method and vector extrapolation can be considered quasi-Newton methods. Irrespective of the name of the method, these methods all have in common that they aim to achieve super-linear convergence in the iterative process, using input/output information only.

1.2.4. Multi-level methods

Multi-grid algorithms are often implemented in flow solvers for efficiency purposes. The multi-grid implementation of the fluid solver can be exploited in order to improve the efficiency of partitioned fluid-structure interaction algorithms without violating the minimal intrusiveness and generality constraints. The efficiency improvement is made by coupling the structure with the fine flow mesh and with the coarse flow mesh, similar to figure 1.3. In this case the high fidelity solver is the discretized system of equations on the fine mesh while the low fidelity solver is constructed from the high fidelity equations using Galerkin projection with restriction (R_f) and prolongation (P_f) operators, see Figure 1.4. Defect-correction is used on the fluid-structure interaction interface to speed up the coupling iterations. These so called *multi-level* methods are highly successful, especially in combination with high-order time integration methods [29, 30].

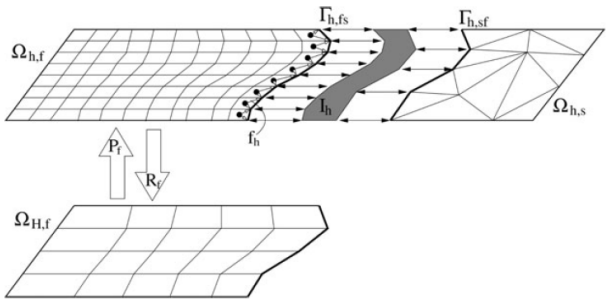


Figure 1.4: Schematic of a multi-level coupling algorithm. Source: Multi-Level Accelerated Sub-Iterations for Fluid-Structure Interaction. Lecture Notes in Computational Science and Engineering Volume 73, 2010, pp 1-25

The idea that multi-level defect-correction can be accelerated with a quasi-Newton scheme was proposed in [47]. An inner-outer iterative scheme is presented with the multi-level method as the inner method and the quasi-Newton method as the outer method. In principle any quasi-Newton method can be selected for the outer scheme, making the method a special case of a much broader class of methods. Not surprisingly, given the super-linear convergence of quasi-Newton methods, a speedup is observed with respect to the multi-level method without quasi-Newton acceleration. It is also possible to improve the Jacobian of a Quasi-Newton method in a multi-level way, see [48]. Accelerations can be achieved with this method when the number of degrees of freedom between the grid levels is sufficient.

1.2.5. Multi-solver methods

A multi-solver method uses more than one flow solver and/or one structure solver for a single unsteady fluid-structure interaction simulation. The acceleration is achieved by running the flow solvers and structure solvers in parallel in a single time-step. How this can be done in a smart and efficient way is presented in [33]. By collecting data from previous time steps in the simulation and recalculating input/output relations for the current time step in parallel, a remarkable efficiency gain can be achieved. For the test case under consideration, the number of coupling iterations decreases with 50% when 8 different flow solvers are used in parallel. The principle of the multi-solver algorithm is shown in Figure 1.5. Since the parallelization of the multi-solver algorithm happens solely on the fluid-structure interface, the speedup is independent on parallelization within the flow solvers. The total computational cost of the additional flow solvers is however large, making it a wise decision to first parallelize a single flow solver in combination with an efficient coupling algorithm, before a multi-solver algorithm is implemented.

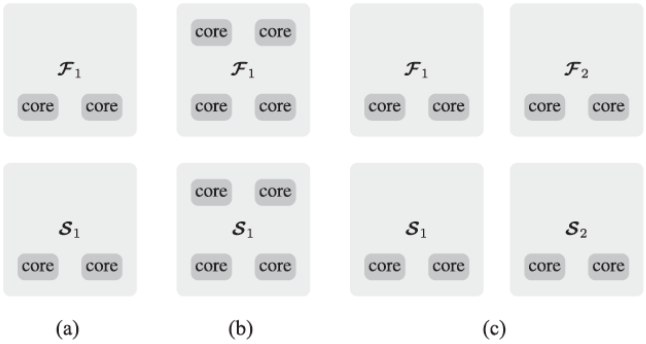


Figure 1.5: Multi-solver principle: (a) fluid and structure solver on two cores. With twice as many cores either (b) the number of cores per solver can be increased or (c) two additional flow solvers can be started. Source: Multi-solver algorithms for the partitioned simulation of fluid-structure interaction. Computer Methods and Applied Mechanics Volume 200, 2011, pp 2195-2210

1.2.6. Multi-fidelity methods

In this dissertation multi-fidelity methods are developed to speed up partitioned fluid-structure interaction simulations. A multi-fidelity algorithm exploits the information of a lower fidelity model to speed up the computations of the high-fidelity model. Using this definition, a multi-level algorithm certainly is a multi-fidelity method. The difference is that multi-fidelity algorithms are a much broader class of coupling algorithms since no predefined choice is made for the low- and high fidelity model. The only requirement is that the low fidelity flow solver is computationally cheap with respect to the more expensive high fidelity flow solver and that it approximates the same physics. Space-mapping [49] can be used to connect the low- and high fidelity solvers, thereby accelerating the iterative process.

space-mapping

The concept of space-mapping was first conceived by J.W. Bandler in 1993. Bandler started to question the concept of ”model” and the recognition of ”real” objects like churches and houses and how these objects could be mapped to an element of a library of preconceived models in one’s brain, see figure 1.6. By scaling , shifting, rotating, twisting and elimination of detail one can establish a mapping between the models.

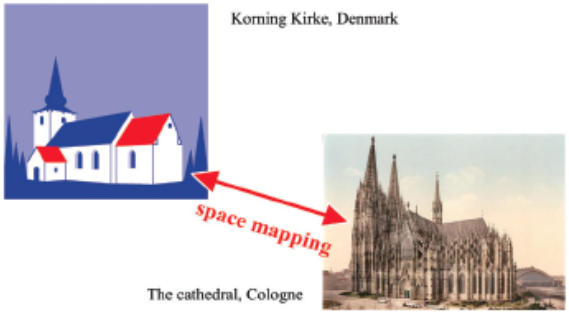


Figure 1.6: The concept of space-mapping. Source: J.W. Bandler "Have you ever wondered about the engineer's mysterious 'feel' for a problem" IEEE Canadian Review, no. 70, pp. 50-60, Summer 2013.

Realizing that this process happens unconsciously, Bandler stated in [50]: "I was searching not for mathematics but for the engineer's 'feel'".

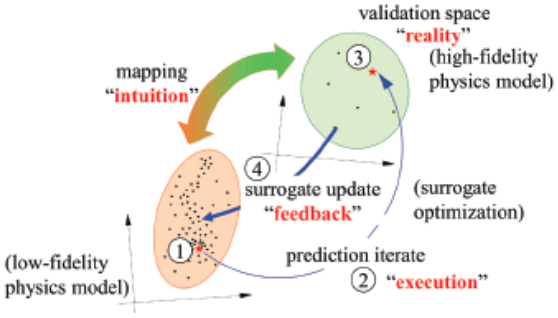


Figure 1.7: The concept of space-mapping. Source: J.W. Bandler "Have you ever wondered about the engineer's mysterious 'feel' for a problem" IEEE Canadian Review, no. 70, pp. 50-60, Summer 2013.

By constructing a low-fidelity model in the brain, intuition can be perceived as the mapping between reality and the low-fidelity model. The low-fidelity model can be corrected iteratively such that a real world objective can be achieved, see figure 1.7. This basic idea lies at the root of the space-mapping technique. By establishing a mapping between two models of different complexity, the iterative process on the high fidelity model can be accelerated. Although space-mapping is currently mainly applied in the field of optimization, it can also be used to efficiently solve the coupled problem at each time step of a partitioned FSI simulation since the underlying principles of space-mapping are quite general. Minimization of the interface residual

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} ||\mathcal{R}(\mathbf{x})||,$$

can be seen as an optimization problem. Depending on how the mapping is used in the iterative process, different space-mapping algorithms emerge. In the following, we introduce Aggressive Space-Mapping, Output Space-Mapping and Manifold-Mapping.

Aggressive space-mapping

A mapping between the input space of the low fidelity model and the input space of the high fidelity model is constructed during the coupling iterations: the space-mapping function. The space-mapping function keeps track of the differences between the high and low fidelity models during the coupling iterations. A-priori knowledge of the exact inverse space-mapping function would allow for the direct computation of the high fidelity solution by the inverse mapping of the low fidelity solution to the high fidelity space. However, such a-priori knowledge is not available. This necessitates the iterative approximation of the inverse space-mapping function during the partitioned coupling iterations. An equivalent optimization problem can be formulated

$$\arg\min_{\mathbf{x}} ||\mathcal{R}(\mathbf{x})|| \leftrightarrow \arg\min_{\mathbf{x}} ||\mathcal{P}(\mathbf{x}) - \mathbf{z}^*||,$$

where $\mathcal{P}(\mathbf{x})$ is the space-mapping function and $\mathbf{z}^*$ is the solution of the low-fidelity model. This is called the *primal formulation*. When the space-mapping function is expanded in a first order Taylor series and when the Jacobian is iteratively approximated using a quasi-Newton method the so called Aggressive Space-Mapping (ASM) algorithm results. The iterative update then becomes

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\frac{\partial \mathcal{P}}{\partial \mathbf{x}}\right)^{-1} \left(\mathcal{P}(\mathbf{x}^k) - \mathbf{z}^*\right).$$

The performance of the ASM method to speed up partitioned FSI simulations is investigated in this dissertation.

Output Space-Mapping

In addition to a mapping between the input space of the low fidelity model and the input space of the high fidelity model it is also possible to make a mapping between the output space of the low fidelity model and the output space of the high fidelity model. First, we formulate the update as

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} ||\mathcal{R}_s^k(\mathbf{x})||, \tag{1.2}$$

where $\mathcal{R}_s^k(\mathbf{x})$ denotes the surrogate model at iterate k. This is called the *dual formulation*. The surrogate model at iterate k is constructed from the high fidelity model using input mappings and output mappings. The reason to introduce an additional output mapping is two-fold. First, it is expected that the algorithm converges faster than ASM since not only the input space of the models is aligned but also the output space. Second, ASM may not always converge due to problems related to model flexibility. By correcting the output of the low fidelity model an algorithm is designed that converges more often to the high fidelity solution. Output space-mapping is for the first time successfully applied to reduce the number of coupling iterations of several fluid-structure interaction problems in [51]

Manifold Mapping

The misalignment between the low-fidelity model and the high-fidelity model can be corrected iteratively using rotation and translation. This process can be seen as the mapping between the low-fidelity (coarse) manifold and the high-fidelity (fine) manifold. Manifold mapping is an improved version of space-mapping that has provable convergence properties, see [52].

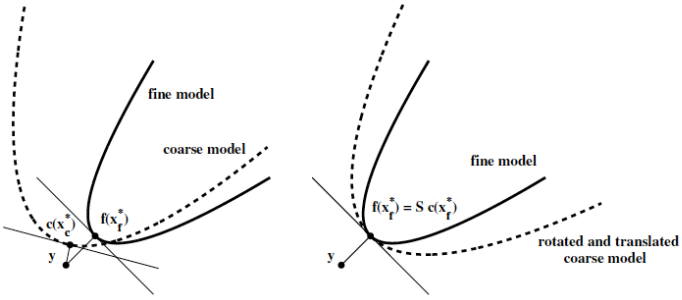


Figure 1.8: The concept of manifold-mapping. Source: Multi-Level Optimization: Space Mapping and Manifold Mapping, Ph. D. thesis, University of Amsterdam (2007).

The concept of manifold-mapping is illustrated in figure 1.8. In this figure, the vector $\mathbf{y}$ denotes the design specification. When manifold mapping is used to accelerate fluid-structure interaction simulations, we have $\mathbf{y} = \mathbf{0}$ and the manifolds are the low fidelity interface residual space and the high fidelity interface residual space respectively. Upon convergence, the low fidelity model is aligned with the high fidelity model. Manifold mapping is for the first time successfully applied in order to reduce the number of coupling iterations in a strong fluid-structure interaction problem in [53].

1.3. Multi-fidelity analysis for uncertainty quantification

Uncertainties can arise from various sources. A frequently used classification of uncertainty is

- Epistemic uncertainty
- Aleatory uncertainty
- Human error

Epistemic uncertainty results from ignorance: the lack of information or understanding of the physics of the problem. Examples are unmodeled nonlinearity, errors in the aerodynamics prediction or lack of information about damping. Epistemic uncertainty can be reduced by collecting more information about the origin of the uncertainty. On the other hand, aleatory uncertainty is the class of uncertainty

that is *irreducible*. Examples of aleatory uncertainty are material imperfections and fabrication errors that result in e.g. geometrical uncertainty. The most common approach for modeling aleatory uncertainty is the probabilistic approach. In this approach uncertainty is modeled using random parameters that follow a predescribed probability density function.

1.3.1. Uncertainty Quantification

In the field of Uncertainty Quantification (UQ), a collection of methods is used that propagate uncertainty through systems. In this dissertation we focus on forward uncertainty propagation: the uncertainty in the system output due to input parametric variability. The output is also called the Quantity of Interest (QoI). The goal of UQ is to obtain statistical moments of the QoI or to obtain the probability that the QoI will exceed a predefined threshold. The latter is the subject of reliability analysis. Random field discretizations are used to express the variability in a finite number of random variables: the random variable space. The number of random variables necessary to describe the variability accurately can however be quite large. Many UQ methods have dimensionality issues: The computational cost increases rapidly with the dimensionality of the problem. This problem is known as the *curse of dimensionality* [19, 54]. It is therefore of utmost importance to account for the scaling of the computational cost with respect to the number of problem dimensions in the development of new UQ methods. A key element in the development of new UQ methods is the efficient computation of meta-models. Meta-models can be used to replace the high fidelity model or to accelerate the analysis on the high fidelity model.

1.3.2. Aim and motivation

Two major problems are observed in the development of computational methods for Uncertainty Quantification. These are the curse of dimensionality and the lack of error estimation techniques when meta-model substitution is used. In order to make a step in the right direction we focus on two aspects of UQ. First, we focus on construction of efficient high dimensional meta-models. Central to this part of the work is the use of the adjoint method. Second, we address how the meta-model can be used to accelerate importance sampling. Central to this part of the work is the estimation of meta-model errors and importance sampling errors. The aim is to investigate the performance of these methods and to identify possible bottlenecks.

1.3.3. Perturbation methods

If the QoI and its gradient w.r.t. the random variables are calculated at a single point in the random variable space, the perturbation method [55] can be used to estimate the statistical moments of interest. A Taylor series approximation is used to construct a linear response surface that is locally accurate. Typically, the gradients are computed using finite differences or the adjoint method.

When the response surface is linear, the mean and variance are estimated using the QoI evaluated at the point of expansion and the inner product of the gradient with itself respectively.

Problems arise when the actual response is highly nonlinear. The linear approximation of the response surface is no longer sufficient in that case. The local approximation of the response can be enhanced using a second order Taylor expansion. However, this requires computation of the diagonal of the Hessian for the improved estimate of the mean and the full Hessian for the improved estimate of the variance. This complicates the use of second order perturbation methods in two ways. First, second order derivatives with respect to parameters are not available in most commercial codes. Second, the number of second order derivatives that needs to be computed scales with the number of random variables squared, which results in a dimensionality problem.

Another complication is the choice of the point of expansion in the Taylor series approximation. It is not clear a priori which point of expansion gives the best approximation to the actual response. Common approaches are mean-centered, median-centered and mode-centered perturbation methods, see [12].

1.3.4. Uncertainty quantification using Kriging

If the QoI and its gradients w.r.t. the random variables are calculated at several points in the random variable space, a wide number of mathematical techniques can be used to obtain a response surface approximation. Response surfaces can be build from observable data using statistical inference techniques. Particularly suitable are methods that use probability statements conditioned on observed data - the so called Bayesian approach, see [56]. These probability statements form a so called stochastic response surface. The stochastic response surfaces can subsequently be used to calculate the statistical moments of the QoI or to obtain the probability of failure as a post-processing step.

A powerful Bayesian inference technique, originating from the field of Geology, is Kriging [57]. The result of Kriging is a stochastic response surface conditioned on observed values of the QoI. The Kriging predictor is given by

$$\mathbb{E}(\mathbf{x}|\mathbf{y}) = \ + \mathbf{P}\mathbf{H}'(\mathbf{R} + \mathbf{H}\mathbf{P}\mathbf{H}')^{-1}(\mathbf{y} - \mathbf{H}),$$

which updates the new values $\mathbf{x}$ conditional on the computed values $\mathbf{y}$, starting from a prior and prior covariance matrix $\mathbf{P}$, while $\mathbf{R}$ contains the observation errors and $\mathbf{H}$ is the observation matrix which selects the computed results from the total set of results. The elements of $\mathbf{P}$ can be constructed from a Gaussian correlation function

$$p_{ij} = \exp\left(-\sum_d \frac{h_{ij,d}^2}{2\theta_d^2}\right),$$

where the lag $h_{ij,d}$ is the distance - in the random variable space - between the results i and j and θ_d is the correlation range in each dimension.

A recent extension of Kriging, named Gradient Enhanced Kriging (GEK) [54], includes the gradients of the QoI to the list of observed values of the QoI, thereby gaining accuracy over ordinary Kriging. Since the QoI and its gradient are calculated at several points in the random variable space, GEK can be considered an extension

of the perturbation method. Instead of using a second order Taylor series expansion at a single point in the random variable space, the approach is now to calculate the QoI and its gradient at a grid of points - the so called Design of Experiment (DoE) - in order to capture the nonlinearity of the response. When the gradients are calculated efficiently - using the adjoint method - GEK scales favorable with the number of dimensions.

1.3.5. Crude Monte Carlo methods

The sampling-based crude Monte Carlo method is the most general method to compute the statistical moments of the QoI. No assumptions regarding the distribution of the QoI are required. Moreover, the required computational effort of the Monte Carlo method does not scale with the number of random variables. In this method, realizations of the random variables are sampled and the deterministic analysis is done for each realization. The mean and standard deviation are subsequently estimated by

$$\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N} QoI(\mathbf{z}_i) \quad \text{and}$$

$$\bar{\sigma} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(QoI(\mathbf{z}_i) - \bar{\mu})^2},$$

where the vector $\mathbf{z}_i$ are the realizations of the random vector $\tilde{\mathbf{z}}$ which contains all randomness in the problem under consideration. The required accuracy is commonly specified by a confidence level C, defined as

$$\Pr\left[\left|\frac{\bar{\mu} - \mu}{\mu}\right| \geq 1 - C\right] \leq 1 - C,$$

for the mean. A similar expression can be written for the standard deviation. Using certain mild assumptions, the number of required samples for the standard deviation is given by

$$N \geq \frac{1}{2}\left(\frac{\Phi^{-1}(\frac{1+C}{2})}{1-C}\right)^2,$$

where Φ denotes the standard normal cumulative density function. For a 95% confidence level of the standard deviation a sample size of 768 is required, for a confidence level of 99% this number increases to 33,174. Clearly, crude Monte Carlo requires too many samples to be useful in practice.

1.4. Breakdown of the dissertation

This dissertation is divided into two parts. Each part consists of a number of test cases in order to assess the performance of the algorithms. The choice for the test cases is most often based on a trade-off between simplicity and demonstrateability. The transonic airfoil problem is however a predefined FFAST test case. The breakdown is as follows:

1. Part I: **Multi-fidelity acceleration of high fidelity fluid-structure interaction simulations**
   (a) **Space-mapping in FSI: theory (Chapter 2):** Theoretical development and explanation of different space-mapping methods.
   (b) **Space-mapping in FSI: Numerical experiments (Chapter 3):** The application of space-mapping in order to reduce the computational effort of FSI simulations. The test cases are:
      i. *The 1-D piston problem:* A compressible inviscid fluid in a closed section which is in interaction with a piston having a mass en stiffness.
      ii. *The 2-D supersonic panel problem:* A flexible panel with a mass and stiffness which is in interaction with a fluid at supersonic speed.
      iii. *The 1-D flexible tube problem:* An incompressible fluid flowing through a flexible tube.
2. Part II: **Multi-fidelity analysis for uncertainty quantification**
   (a) **High-dimensional meta-models for UQ (Chapter 4):** The application of Gradient Enhanced Kriging and the efficient use of the adjoint method to mitigate the curse of dimensionality. The test cases are:
   (b) **The panel divergence problem (Chapter 5):** The fluid-structure stability boundary is the quantity of interest in this test case. Parametric uncertainty enters the problem via the random stiffness of the panel. The aim is to obtain an accurate meta-model of the divergence Mach number
   (c) **The transonic airfoil problem (Chapter 6):** The airfoil drag is the quantity of interest in this test case. Parametric uncertainty enters the problem via the random shape deformation of the airfoil. The aim is to obtain the statistical moments of the airfoil drag, based on the meta-model.

The dissertation ends with conclusions and recommendations in Chapter 7.

1

References

[1] D. Jones and A. Gaitonde, *Future Fast Methods for Loads Calculations: The 'FFAST' project,* in *Innovation for sustainable aviation in a global environment*, edited by D. Knörzer and J. Szodruch (IOS Press BV, 2012) pp. 110–115.

[2] J. B. Vos, A. Rizzi, D. Darracq, and E. H. Hirschel, *Navier-Stokes solvers in European aircraft design,* Progress in Aerospace Sciences **38**, 601 (2002).

[3] H. Haddad Khodaparast, *Efficient worst case 1-cosine gust loads prediction,* Journal of Aeroelasticity and Structural Dynamics **2**, 33 (2012).

[4] C. L. Karr, T. A. Zeiler, and R. Mehrotra, *Determining worst-case gust loads on aircraft structures using an evolutionary algorithm,* Applied Intelligence **20**, 135 (2004).

[5] H. H. Khodaparast, G. Georgiou, J. E. Cooper, L. Travaglini, S. Ricci, G. A. Vio, and P. Denner, *Rapid Prediction of Worst-Case Gust Loads,* 52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference (2011).

[6] H. H. Khodaparast and J. E. Cooper, *Rapid Prediction of Worst-Case Gust Loads Following Structural Modification,* AIAA Journal **52**, 242 (2014).

[7] A. G. B. Mowat, A. G. Malan, L. H. van Zyl, and J. P. Meyer, *A hybrid finite-volume-ROM approach to non-linear aerospace fluid-structure modelling,* Paper IFASD-2011-08 (2011).

[8] J. Vierendeels, L. Lanoye, J. Degroote, and P. Verdonck, *Implicit coupling of partitioned fluid-structure interaction problems with reduced order models,* Computers & Structures **85**, 970 (2007), fourth MIT Conference on Computational Fluid and Solid Mechanics.

[9] L. Cavagna, S. Ricci, and L. Riccobene, *FAST-GLD: a Fast Tool for the Prediction of Worst Case Gust Loads Based on Neural Networks,* 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference (2013).

[10] R. Ghanem and P. Spanos, *Stochastic Finite Elements: A Spectral Approach* (Springer Verlag, 1991) (reissued by Dover Publications, 2004).

[11] N. Wiener, *The homogeneous chaos,* American Journal of Mathematics **60**, pp. 897 (1938).

[12] C. V. Verhoosel, T. P. Scholcz, and S. J. Hulshoff, *Uncertainty and reliability analysis of fluid–structure stability boundaries,* AIAA Journal **47**, 91 (2009).

[13] C. V. Verhoosel, T. P. Scholcz, S. J. Hulshoff, and M. A. Gutiérrez, *Transient stability analysis of an aeroelastic problem with random fluid and structure properties,* in *Proceedings of the 10th International Conference on Structural*

Safety and Reliability (ICOSSAR10) (2009) pp. 377–384, 13-17 September 2009, Osaka, Japan.

[14] H. H. Khodaparast, J. E. Mottershead, and K. J. Badcock, *Propagation of structural uncertainty to linear aeroelastic stability,* Computers & Structures **88**, 223 (2010).

[15] S. Marques, K. J. Badcock, H. H. Khodaparast, and J. E. Mottershead, *Transonic aeroelastic stability predictions under the influence of structural variability,* Journal of Aircraft **47**, 1229 (2010).

[16] B. Sudret, *Meta-models for structural reliability and uncertainty quantification,* Proc. 5th Asian-Pacific Symp. Struct. Reliab. (APSSRA 2012) Singapore (2012).

[17] B. Rosenbaum and V. Schulz, *Response surface methods for efficient aerodynamic surrogate models,* in *Computational Flight Testing*, Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Vol. 123, edited by N. Kroll, R. Radespiel, J. W. Burg, and K. Sørensen (Springer Berlin Heidelberg, 2013) pp. 113–129.

[18] T. West, B. Reuter, E. Walker, W. L. Kleb, and M. A. Park, *Uncertainty quantification and certification prediction of low-boom supersonic aircraft configurations,* in *32nd AIAA Applied Aerodynamics Conference* (American Institute of Aeronautics and Astronautics, 2014).

[19] J. A. S. Witteveen and H. Bijl, *Efficient quantification of the effect of uncertainties in advection-diffusion problems using polynomial chaos,* Numerical Heat Transfer, Part B: Fundamentals **53**, 437 (2008).

[20] V. J. Romero, R. Slepoy, L. P. Swiler, A. A. Giunta, and T. Krishnamurthy, *Error estimation approaches for progressive response surfaces-more results,* in *Proceedings of the international modal analysis conference*, 2 (2006) pp. 782–802.

[21] M. Meckesheimer, A. J. Booker, R. R. Barton, and T. W. Simpson, *Computationally inexpensive metamodel assessment strategies,* AIAA Journal **40**, 2053 (2002).

[22] T. Goel, R. Hafkta, and W. Shyy, *Comparing error estimation measures for polynomial and kriging approximation of noise-free functions,* Structural and Multidisciplinary Optimization **38**, 429 (2009).

[23] V. Dubourg, B. Sudret, and F. Deheeger, *Metamodel-based importance sampling for structural reliability analysis,* Probabilistic Engineering Mechanics **33**, 47 (2013).

[24] B. Echard, N. Gayton, and M. Lemaire, *Ak-mcs: An active learning reliability method combining kriging and monte carlo simulation,* Structural Safety **33**, 145 (2011).

[25] Y. Bazilevs, J. R. Gohean, T. J. R. Hughes, R. D. Moser, and Y. Zhang, *Patient-specific isogeometric fluid–structure interaction analysis of thoracic aortic blood flow due to implantation of the jarvik 2000 left ventricular assist device,* Computer Methods in Applied Mechanics and Engineering **198**, 3534 (2009), models and Methods in Computational Vascular and Cardiovascular Mechanics.

[26] C. M. Scotti and E. A. Finol, *Compliant biomechanics of abdominal aortic aneurysms: A fluid-structure interaction study,* Computers and Structures **85**, 1097 (2007), fourth MIT Conference on Computational Fluid and Solid Mechanics.

[27] P. J. Schmid and E. de Langre, *Transient growth before coupled-mode flutter,* Journal of Applied Mechanics **70**, 894 (2003).

[28] A. D. Ronch, K. J. Badcock, A. Khrabrov, M. Ghoreyshi, and R. Cummings, *Modeling of unsteady aerodynamic loads,* in *AIAA Atmospheric Flight Mechanics Conference* (American Institute of Aeronautics and Astronautics, 2011).

[29] A. H. Zuijlen, S. Bosscher, and H. Bijl, *Two level algorithms for partitioned fluid-structure interaction computations,* Computer Methods in Applied Mechanics and Engineering **196**, 1458 (2007).

[30] A. H. Zuijlen and H. Bijl, *Multi-level acceleration for sub-iterations in partitioned fluid-structure interaction,* AIP Conference Proceedings **1168**, 1347 (2009).

[31] E. van Brummelen, K. van der Zee, and R. de Borst, *Space/time multigrid for a fluid-structure-interaction problem,* Applied Numerical Mathematics **58**, 1951 (2008).

[32] J. Hron and S. Turek, *A monolithic fem/multigrid solver for an ale formulation of fluid-structure interaction with applications in biomechanics,* in *Fluid-Structure Interaction*, Lecture Notes in Computational Science and Engineering, Vol. 53, edited by H.-J. Bungartz and M. Schäfer (Springer Berlin Heidelberg, 2006) pp. 146–170.

[33] J. Degroote and J. Vierendeels, *Multi-solver algorithms for the partitioned simulation of fluid–structure interaction,* Computer Methods in Applied Mechanics and Engineering **200**, 2195 (2011).

[34] C. Michler, H. van Brummelen, and R. de Borst, *An investigation of interface-gmres(r) for fluid–structure interaction problems with flutter and divergence,* Computational Mechanics **47**, 17 (2011).

[35] C. Michler, E. H. van Brummelen, and R. de Borst, *An interface newton-krylov solver for fluid-structure interaction,* International Journal for Numerical Methods in Fluids **47**, 1189 (2005).

[36] U. Küttler and W. Wall, *Fixed-point fluid-structure interaction solvers with dynamic relaxation,* Computational Mechanics **43**, 61 (2008).

[37] U. Küttler and W. A. Wall, *Vector extrapolation for strong coupling fluid-structure interaction solvers,* Journal of Applied Mechanics **76**, 021205 (2009).

[38] J. Degroote, K.-J. Bathe, and J. Vierendeels, *Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction,* Computers and Structures **87**, 793 (2009), fifth MIT Conference on Computational Fluid and Solid Mechanics.

[39] R. Haelterman, J. Degroote, D. v. Heule, and J. Vierendeels, *The quasi-newton least squares method: A new and fast secant method analyzed for linear systems,* SIAM J. Numer. Anal. **47**, 2347 (2009).

[40] T. Gallinger and K. Bletzinger, *Comparison of algorithms for strongly coupled partitioned fluid-structure interaction,* in *Proceedings of the 5th European Conference on Computational Fluid Dynamics (ECCOMAS CFD 2010)* (2010) 14-17 June 2010, Lisbon, Portugal.

[41] R. Haelterman, J. Degroote, D. van Heule, and J. Vierendeels, *On the similarities between the quasi-newton inverse least squares method and gmres,* SIAM Journal on Numerical Analysis **47**, 4660 (2010).

[42] S. Badia, A. Quaini, and A. Quarteroni, *Modular vs. non-modular preconditioners for fluid-structure systems with large added-mass effect,* Computer Methods in Applied Mechanics and Engineering **197**, 4216 (2008).

[43] E. H. van Brummelen, *Partitioned iterative solution methods for fluid-structure interaction,* International Journal for Numerical Methods in Fluids **65**, 3 (2011).

[44] G. P. Guruswamy, *Integrated approach for active coupling of structures and fluids,* AIAA Journal **27**, 788 (1989).

[45] L. Djayapertapa and C. B. Allen, *Simulation of transonic flutter and active shockwave control,* International Journal of Numerical Methods for Heat & Fluid Flow **14**, 413 (2004).

[46] S. Badia and R. Codina, *On some fluid-structure iterative algorithms using pressure segregation methods. Application to aeroelasticity,* International Journal for Numerical Methods in Engineering **72** (2007).

[47] J. J. Kreeft, M. Weghs, A. H. Zuijlen, and H. Bijl, *Multi-level and Quasi-Newton acceleration for strongly coupled partitioned fluid-structure interaction,* in *Proceedings of the 4th International Conference on Computational Methods for Coupled Problems in Science and Engineering (ECCOMAS 2011)* (2011) 20-22 June 2011, Kos Island, Greece.

[48] J. Degroote and J. Vierendeels, *Multi-level quasi-newton coupling algorithms for the partitioned simulation of fluid-structure interaction,* Computer Methods in Applied Mechanics and Engineering **225-228**, 14 (2012).

[49] D. Echeverría and P. W. Hemker, *Space mapping and defect correction,* Computational Methods in Applied Mathematics **5**, 107 (2005).

[50] J. W. Bandler, *"Have your ever wondered about the engineer's mysterious 'feel' for a problem",* IEEE Canadian Review , 50 (Summer 2013).

[51] L. Florentie, *Acceleration of partitioned fluid-structure interaction simulations by means of space mapping,* Master thesis (2012).

[52] D. Echeverria, *Multi-level optimization. Space mapping and manifold mapping*, Ph.D. thesis, University of Amsterdam (2007).

[53] D. S. Blom, A. van Zuijlen, and H. Bijl, *Acceleration of strongly coupled fluid-structure interaction with manifold mapping,* in *Proceedings of the 11th Wold Congress on Ccomputational Mmechanics (WCCM XI)* (2014) 20-25 July 2014, Barcelona, Spain.

[54] J. H. S. de Baar, R. P. Dwight, and H. Bijl, *Improvements to gradient-enhanced kriging using a bayesian interpretation,* International Journal for Uncertainty Quantification **4**, 205 (2014).

[55] M. A. Gutierrez and S. Krenk, *Encyclopedia of Computational Mechanics* (Wiley, New York, 2004) pp. 657–681.

[56] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science)*, 2nd ed. (Chapman and Hall/CRC, 2003).

[57] N. Cressie, *The origins of kriging,* Mathematical Geology **22**, 239 (1990).

I

Multi-fidelity acceleration of high fidelity fluid-structure simulations

2

Space-mapping in FSI: Theory

In this dissertation the use of low fidelity models to speed up partitioned coupling simulations applied to high fidelity models is investigated, the so called *multi-fidelity* approach [1]. Without loss of generality it is assumed that two solvers are available: a computationally cheap low fidelity fluid solver and a computationally expensive high fidelity fluid solver. A mapping is defined between the input space of the low fidelity model and the input space of the high fidelity model during the coupling iterations: the space-mapping function. A-priori knowledge of the exact inverse space-mapping function would allow for the direct computation of the high fidelity solution by the inverse mapping of the low fidelity solution to the high fidelity space. However, such a-priori knowledge is not available. This necessitates the iterative approximation of the inverse space-mapping function during the partitioned coupling iterations. The space-mapping function keeps track of the differences between the high and low fidelity models during the coupling iterations and this information subsequently used to speedup the computations. When the space-mapping function is expanded in a first order Taylor series and when the Jacobian is iteratively approximated using input/output information the so called Aggressive Space-Mapping (ASM) algorithm results. Although space-mapping [1] is currently mainly applied in the field of optimization, it can also be used to efficiently solve the coupled problem at each time step of a partitioned FSI simulation since the underlying principles of space-mapping are quite general.

[1] Chapter 2 and Chapter 3 are based on the journal paper: **T.P. Scholcz, A.H. van Zuijlen and H. Bijl**, *Space-mapping in fluid-structure interaction problems*, Computer Methods in Applied Mechanics and Engineering **281** (2014).

2.1. Problem formulation

A typical FSI model consists of a fluid model defined on a deformable domain Ω_f – numerically implemented using the Arbitrary Langrangian Eulerian (ALE) formulation [2] – which is in interaction with a structure model defined on Ω_s and modeled using the Lagrangian formulation. The fluid domain and structure domain both have a fluid-structure interaction interface Γ_I^f and Γ_I^s which are identical, such that both domains are coupled [3]. A discretization technique is used to obtain the semi-discrete systems of equations describing the dynamics of each physical system. Typically the Finite Volume Method (FVM) is used to obtain the system describing the fluid dynamics whereas the Finite Element Method (FEM) is used to obtain the structural system equations. Let the vector $\mathbf{v}$ denote the discrete state vector of the fluid in Ω_f and vector $\mathbf{u}$ the the discrete state vector of the structure in Ω_s at the new time level t^{n+1}. Hiding the dependency on the solution of previous time levels, the coupled problem at time step t^{n+1} is formulated as [4]

$$\mathbf{r}^f(\mathbf{v}; \xi_x(\mathbf{u})) = \mathbf{0} \quad \mathbf{v} \in \mathbb{R}^{N_f} \tag{2.1}$$

$$\mathbf{r}^s(\mathbf{u}; \xi_y(\mathbf{v})) = \mathbf{0} \quad \mathbf{u} \in \mathbb{R}^{N_s}. \tag{2.2}$$

Here, $\mathbf{r}^f$ denotes the residual of the discrete fluid equations and $\mathbf{r}^s$ the residual of the discrete structure equations. The function $\xi_x : \mathbb{R}^{N_s} \to \mathbb{R}^{N_I^s}$ maps the structural state vector $\mathbf{u} \in \mathbb{R}^{N_s}$ to the interface displacement vector $\mathbf{x} \in \mathbb{R}^{N_I^s}$, hence $\mathbf{x} = \xi_x(\mathbf{u})$. The function $\xi_y : \mathbb{R}^{N_f} \to \mathbb{R}^{N_I^f}$ maps the fluid state vector $\mathbf{v} \in \mathbb{R}^{N_f}$ to the interface pressure $\mathbf{y} \in \mathbb{R}^{N_I^f}$, hence $\mathbf{y} = \xi_y(\mathbf{v})$. Given a certain interface displacement vector $\mathbf{x}$

$$\mathbf{y} = \xi_y(\mathbf{v}) \quad \text{for} \quad \mathbf{v} = \arg \min_{\mathbf{v} \in \mathbb{R}^{N_f}} ||\mathbf{r}^f(\mathbf{v}; \mathbf{x})||, \tag{2.3}$$

defines the *high fidelity fluid operator* $\mathcal{F} : \mathbb{R}^{N_I^s} \to \mathbb{R}^{N_I^f}$

$$\mathbf{y} = \mathcal{F}(\mathbf{x}). \tag{2.4}$$

Numerical evaluation of the high fidelity fluid operator requires the solution of the minimization problem in Eq. (2.3) up to a certain tolerance ϵ_f and the evaluation of the map ξ_y to find the interface pressure from the fluid state vector. The minimization of $\mathbf{r}^f$ is in general performed with a Computational Fluid Dynamics solver. Likewise, given an interface pressure $\mathbf{y}$

$$\mathbf{x} = \xi_x(\mathbf{u}) \quad \text{for} \quad \mathbf{u} = \arg \min_{\mathbf{u} \in \mathbb{R}^{N_s}} ||\mathbf{r}^s(\mathbf{u}; \mathbf{y})||, \tag{2.5}$$

defines the *structure operator* $\mathcal{S} : \mathbb{R}^{N_I^f} \to \mathbb{R}^{N_I^s}$

$$\mathbf{x} = \mathcal{S}(\mathbf{y}). \tag{2.6}$$

Numerical evaluation of the structure operator requires the solution of the minimization problem in Eq. (2.5) up to a certain tolerance ϵ_s and evaluation of the map ξ_x to find the interface displacement from the structure variables. The minimization of $\mathbf{r}^s$ is in general performed with a Computational Structural Dynamics solver. Due to the large range of important scales present in the fluid dynamics it holds in general that $N_f >> N_s$. Evaluation of the fluid operator is computationally much more expensive than evaluation of the structure operator as a consequence.

Continuity of the interface displacement/velocity and force equilibrium on the fluid-structure interface is required in order to satisfy kinematic and dynamic interface conditions on the fluid-structure interaction interface at every time step of a simulation. These conditions are satisfied when [4–6]

$$\mathcal{R}(\mathbf{x}) = \mathbf{0} \quad \text{with} \quad \mathcal{R}(\mathbf{x}) = \mathcal{S} \circ \mathcal{F}(\mathbf{x}) - \mathbf{x}, \tag{2.7}$$

where $\mathcal{R} : \mathbb{R}^{N_I^S} \rightarrow \mathbb{R}^{N_I^S}$ is the *interface residual* function. Strong coupling algorithms aim to minimize the interface residual $\mathcal{R}$ to a certain tolerance ϵ_I using a minimum number of (expensive) fluid operator evaluations:

$$\mathbf{x}^* = \arg \min_{\mathbf{x} \in \mathbb{R}^{N_I^S}} ||\mathcal{R}(\mathbf{x})||. \tag{2.8}$$

Note that when $\epsilon_f = \epsilon_s = \epsilon_I = 0$ it holds that $\mathbf{u} = \mathbf{u}^*$ and $\mathbf{v} = \mathbf{v}^*$ satisfying Eq. (2.1) and (2.2) and the unique interface displacement and pressure are found from $\mathbf{x}^* = \xi_x(\mathbf{u}^*)$ and $\mathbf{y}^* = \xi_y(\mathbf{v}^*)$ respectively. The most basic strong coupling algorithm is the fixed point iteration scheme given in algorithm 1.

Algorithm 1 Fixed point iteration scheme

1: $k = 0$
2: $\mathbf{r}^k = \mathcal{R}(\mathbf{x}^k)$
3: **while** $||\mathbf{r}^k|| > \epsilon_I$ **do**
4: $\quad \mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{r}^k$
5: $\quad k = k + 1$
6: $\quad \mathbf{r}^k = \mathcal{R}(\mathbf{x}^k)$
7: **end while**

Algorithm 1 may converge slowly and has a poor robustness, see [5, 7]. To obtain better performance, so called Quasi-Newton algorithms were developed, [6–12]. Quasi-Newton methods are more robust and may obtain superlinear convergence, see [9, 12].

2.2. The Quasi-Newton Inverse Least Squares algorithm

The purpose of this section is two-fold. It will introduce the Quasi-Newton Inverse Least Squares (QN-ILS) method as an accelerator of the fixed point iteration scheme given in algorithm 1 and provide some results which are necessary for the development of the Aggressive Space Mapping algorithm in section 2.3.2.

Introducing $\mathcal{H} = \mathcal{S} \circ \mathcal{F}$, the fixed point iteration update on line 4 of algorithm 1 can be written as

$$\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^k). \tag{2.9}$$

To improve convergence of the fixed point iteration scheme $\mathbf{x}^k$ is replaced by a better candidate $\mathbf{x}^{new}$ such that

$$\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^{new}). \tag{2.10}$$

It is assumed that the new candidate can be written as a linear combination of the previous iterates $\mathbf{x}^0 ... \mathbf{x}^k$

$$\mathbf{x}^{new} \in \mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k). \tag{2.11}$$

The new candidate is substituted into the expression for the residual

$$\mathcal{R}(\mathbf{x}^{new}) = \mathcal{R}(\mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k)), \tag{2.12}$$

and subsequently linearized

$$\mathcal{R}(\mathbf{x}^{new}) \approx \mathcal{R}(\mathbf{x}^k) + \left(\frac{\partial \mathcal{R}}{\partial \mathbf{x}}\right) \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k). \tag{2.13}$$

Equation (2.13) can be further simplified to

$$\mathcal{R}(\mathbf{x}^{new}) \approx \mathbf{r}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{r}^i - \mathbf{r}^k), \tag{2.14}$$

where $\mathbf{r}^i = \mathcal{R}(\mathbf{x}^i)$ for $i = 0 ... k$ are available from previous iterates. The coefficients c_i^k are then found from a minimization of the linearized residual $\mathcal{R}(\mathbf{x}^{new})$

$$\mathbf{c}^k = \arg \min_{\mathbf{c}^k \in \mathbb{R}^k} ||\mathbf{r}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{r}^i - \mathbf{r}^k)||, \tag{2.15}$$

such that the better candidate is found from

$$\mathbf{x}^{new} = \mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k). \tag{2.16}$$

Substituting $\mathbf{x}^{new}$ from Eq. (2.16) in the expression for the update in Eq. (2.10) it is found that

$$\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k)) \tag{2.17}$$

$$\approx \mathcal{H}(\mathbf{x}^k) + \left(\frac{\partial \mathcal{H}}{\partial \mathbf{x}}\right) \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k) \tag{2.18}$$

$$\approx \mathbf{x}_{\mathcal{H}}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k), \tag{2.19}$$

where $\mathbf{x}_{\mathcal{H}}^i = \mathcal{H}(\mathbf{x}^i)$ for $i = 0 \ldots k$. If the update is computed as $\mathbf{x}^{k+1} = \mathbf{x}^{new}$ instead of $\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^{new})$ the algorithm brakes down since the new input for $\mathcal{H}$ would be a pure linear combination of previous inputs, see [13]. In [14] the update is computed as $\mathbf{x}^{k+1} = f(\mathbf{x}^{new}, \mathcal{H}(\mathbf{x}^{new}), \beta_k)$, where β_k are *relaxation* parameters for each iterate k. In this algorithm, the QN-ILS algorithm results as a special case for $\beta_k = 1$.

The linearization in Eq. (2.19) avoids a true evaluation of $\mathcal{H}(\mathbf{x}^{new})$ since all the displacement vectors in Eq. (2.19) are readily available when they are stored in previous iterations.

Practical implementation details of the QN-ILS method are found in [6]. The differences in Eq. (2.14) and Eq. (2.19) are calculated as

$$\Delta \mathbf{r}^i = \mathbf{r}^i - \mathbf{r}^k \tag{2.20}$$

$$\Delta \mathbf{x}_{\mathcal{H}} = \mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k, \tag{2.21}$$

for $i = 0 \ldots k-1$. Subsequently the differences are stored in separate matrices $\mathbf{V}^k$ and $\mathbf{W}^k$

$$\mathbf{V}^k = [\Delta \mathbf{r}^{k-1}\ \Delta \mathbf{r}^{k-2} \ldots \Delta \mathbf{r}^0] \tag{2.22}$$

$$\mathbf{W}^k = [\Delta \mathbf{x}_{\mathcal{H}}^{k-1}\ \Delta \mathbf{x}_{\mathcal{H}}^{k-2}\ \ldots \Delta \mathbf{x}_{\mathcal{H}}^0]. \tag{2.23}$$

Since $\mathbf{x}_{\mathcal{H}}^k = \mathbf{x}^k + \mathbf{r}^k$ the update in Eq. (2.19) can be written

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{W}^k \mathbf{c}^k + \mathbf{r}^k, \tag{2.24}$$

which is interpreted as a Quasi-Newton method

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\frac{\partial \mathcal{R}^k}{\partial \mathbf{x}}\right)^{-1} \mathbf{r}^k, \tag{2.25}$$

with the inverse of the Jacobian approximated by

$$\left(\frac{\partial \mathcal{R}^k}{\partial \mathbf{x}}\right)^{-1} \approx \mathbf{W}^k \mathbf{R}^{k^{-1}} \mathbf{Q}^{kT} - \mathbf{I}, \tag{2.26}$$

and $\mathbf{Q}^k$ and $\mathbf{R}^k$ the matrices that result from the economy-size QR-decomposition of $\mathbf{V}^k$. The QR-decomposition is used to minimize (2.15) in a least squares sense. The QN-ILS algorithm is summarized in Algorithm 2.

Algorithm 2 QN-ILS

$k = 0$
$\mathbf{x}_{\mathcal{H}}^k = \mathcal{H}(\mathbf{x}^k)$
$\mathbf{r}^k = \mathbf{x}_{\mathcal{H}}^k - \mathbf{x}^k$
while $||\mathbf{r}^k|| > \epsilon_I$ **do**
  if $k = 0$ **then**
    $\mathbf{x}^{k+1} = \mathbf{x}^k - (\nabla_{\mathbf{x}}^k \mathcal{R})^{-1} \mathbf{r}^k$
  else
    for $i = 0$ to $k - 1$ **do**
      $\Delta \mathbf{x}_{\mathcal{H}}^i = \mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k$
      $\Delta \mathbf{r}^i = \mathbf{r}^i - \mathbf{r}^k$
    end for
    $\mathbf{V}^k = \left[\Delta \mathbf{r}^{k-1} \; \Delta \mathbf{r}^{k-2} \; ... \; \Delta \mathbf{r}^0\right]$
    $\mathbf{W}^k = \left[\Delta \mathbf{x}_{\mathcal{H}}^{k-1} \; \Delta \mathbf{x}_{\mathcal{H}}^{k-2} \; ... \; \Delta \mathbf{x}_{\mathcal{H}}^0\right]$
    Calculate $\mathbf{V}^k = \mathbf{Q}^k \mathbf{R}^k$
    Calculate $\mathbf{R}^k \mathbf{c}^k = -\mathbf{Q}^{kT} \mathbf{r}^k$
    $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{W}^k \mathbf{c}^k + \mathbf{r}^k$
  end if
  $k = k + 1$
  $\mathbf{x}_{\mathcal{H}}^k = S \circ F(\mathbf{x}^k)$
  $\mathbf{r}^k = \mathbf{x}_{\mathcal{H}}^k - \mathbf{x}^k$
end while

2.3. Space-mapping methods

In order to enhance the convergence and robustness properties of the QN-ILS algorithm information of simplified fluid operators is included that are much cheaper to evaluate. Simplified fluid operators were used to approximate the Jacobian in the Newton-GMRES method, see [9, 11]. This requires two evaluations of the simplified fluid operator in each Krylov iteration to compute a finite difference approximation of a Jacobian vector product. The simplified fluid operators should be chosen in such a way that the relevant physics of the problem is maintained, neglecting for example certain nonlinearities and/or the deformation of the fluid domain. Once the simplified fluid operator is available it is used as a black-box, in the same fashion as the original fluid operator. In this work a simplified fluid operator is used to find

a preconditioned rootfinding problem, to which an *outer* Quasi-Newton method is applied. A straightforward way to obtain a preconditioned root-finding problem is to apply a transformation of the original rootfinding problem [15]

$$\mathcal{G}(\mathcal{R}(\mathbf{x})) = \mathbf{0}, \tag{2.27}$$

such that

$$\frac{\partial \mathcal{G}}{\partial \mathbf{v}} \frac{\partial \mathcal{R}}{\partial \mathbf{x}} \approx \mathbf{I}, \tag{2.28}$$

if $\mathcal{G}$ is close to $\mathcal{R}^{-1}$. In that case only a few iterations are required to achieve convergence. When the functions $\mathcal{G}$ and $\mathcal{R}$ are not explicitly known and the Jacobians $\frac{\partial \mathcal{G}}{\partial \mathbf{v}}$ and $\frac{\partial \mathcal{R}}{\partial \mathbf{x}}$ are either unavailable or very expensive to compute space-mapping algorithms [1] can be used to find a preconditioned rootfinding problem indirectly. Space-mapping algorithms exploit the combination of less sophisticated models with the accuracy of more complex models, using input/output information only.

Let $\widetilde{\mathbf{r}}^f(\tilde{\mathbf{v}}, \mathbf{z})$ denote the residual of the discrete fluid equations describing the simplified fluid model. The simplified fluid model has the interface displacement $\mathbf{z} \in \mathbb{R}^{\widetilde{N}_I^S}$ as an argument. Given an interface displacement $\mathbf{z}$

$$\tilde{\mathbf{y}} = \xi_p(\tilde{\mathbf{v}}) \quad \text{for} \quad \tilde{\mathbf{v}} = \arg \min_{\tilde{\mathbf{v}} \in \mathbb{R}^{\widetilde{N}_f}} ||\widetilde{\mathbf{r}}^f(\tilde{\mathbf{v}}; \mathbf{z})|| \tag{2.29}$$

defines the *low fidelity fluid operator* $\tilde{\mathbf{y}} = \widetilde{F}(\mathbf{z})$, such that the low fidelity fluid-structure interaction problem becomes

$$\widetilde{\mathcal{R}}(\mathbf{z}) = \mathbf{0} \quad \text{with} \quad \widetilde{\mathcal{R}}(\mathbf{z}) = \mathcal{S} \circ \widetilde{\mathcal{F}}(\mathbf{z}) - \mathbf{z}. \tag{2.30}$$

Note that the state variables of the high fidelity and low fidelity fluid models may differ and that the low fidelity fluid state vector $\tilde{\mathbf{v}}$ is not necessary an element of the same discrete space as the high fidelity fluid state vector $\mathbf{v}$: $\widetilde{N}_f \neq N_f$. Also, on the interface it might happen that $\widetilde{N}_I^f \neq N_I^f$ and $\widetilde{N}_I^S \neq N_I^S$. In that case interface restriction and prolongation operators can be used in the space-mapping function in order to make the mappings between the two discrete spaces. The solution of the low fidelity fluid-structure interaction problem is given by

$$\mathbf{z}^* = \arg \min_{\mathbf{z} \in \mathbb{R}^{\widetilde{N}_I^S}} ||\widetilde{\mathcal{R}}(\mathbf{z})||. \tag{2.31}$$

The minimization in (2.31) is up to a tolerance $\widetilde{\epsilon}_I$. The "$\arg\min$" notation in (2.31) is used to point out that a partitioned fluid-structure interaction problem is solved without specifying which coupling algorithm is used for the solve since this is not relevant for the understanding of the space mapping method. This notation is used throughout this chapter and when numerical experiments are discussed the choices made for the coupling algorithms are mentioned. The space-mapping approach requires the definition of a space-mapping function. This is the topic of section 2.3.1.

2.3.1. Space-mapping function

A misalignment function between the high fidelity interface residual and the low fidelity interface residual on the fluid-structure interface is defined as

$$r(\mathbf{z}, \mathbf{x}) = ||\widetilde{\mathcal{R}}(\mathbf{z}) - \mathcal{R}(\mathbf{x})||. \tag{2.32}$$

For a given high fidelity model interface displacement $\mathbf{x} \in X$ it is useful to know which low fidelity model interface displacement $\mathbf{z} \in Z$ yields the best approximation to the interface residual $\mathcal{R}$, hence with the smallest misalignment r. Finding the best $\mathbf{z}$ for a given $\mathbf{x}$ defines the space-mapping function $\mathcal{P} : \mathbb{R}^{N_I^s} \rightarrow \mathbb{R}^{N_I^s}$

$$\mathbf{p} = \mathcal{P}(\mathbf{x}) = \arg \min_{\mathbf{z} \in \mathbb{R}^{N_I}} r(\mathbf{z}, \mathbf{x}). \tag{2.33}$$

To evaluate the space-mapping function numerically a second "auxiliary" fluid-structure interaction problem needs to be solved with the low fidelity fluid operator. This problem can be solved with any coupling algorithm as long as it results in a stable and convergent iterative process. The choice for this coupling method defines the *inner* method. An example of the numerical evaluation of the space-mapping function is summarized in Algorithm 3 which uses basic fixed-point iterations as an inner method.

Algorithm 3 Numerical evaluation of $\mathbf{p}^k = \mathcal{P}(\mathbf{x}^k)$

Require: $\mathbf{x}^k, \mathbf{z}^0, \epsilon_s$
1: $i = 0$
2: $\mathbf{r}^k = \mathcal{R}(\mathbf{x}^k)$
3: $\widetilde{\mathbf{r}}^i = \widetilde{\mathcal{R}}(\mathbf{z}^i)$
4: **while** $||\widetilde{\mathbf{r}}^i - \mathbf{r}^k|| > \epsilon_s$ **do**
5: $\quad \mathbf{z}^{i+1} = \mathcal{S} \circ \widetilde{\mathcal{F}}(\mathbf{z}^i) - \mathbf{r}^k$
6: $\quad i = i + 1$
7: $\quad \widetilde{\mathbf{r}}^i = \mathcal{S} \circ \widetilde{\mathcal{F}}(\mathbf{z}^i) - \mathbf{z}^i$
8: **end while**
9: $\mathbf{p}^k = \mathbf{z}^i$
10: **return** $\mathbf{p}^k, \mathbf{r}^k$

Evaluation of the space-mapping function requires a single expensive evaluation of the high fidelity fluid operator and several cheap evaluations of the low fidelity fluid operator to perform the mapping.

2.3.2. The Aggressive Space-Mapping algorithm

The following definition is cited from [1]

Definition 1 *A space-mapping function $\mathcal{P}$ is called a perfect mapping iff $\mathbf{z}^* = \mathcal{P}(\mathbf{x}^*)$.*

Substituting $\mathbf{x}^*$ into the space-mapping function defined by Eq. (2.33) and using the definition in Eq. (2.32), Eq. (2.8) and Eq. (2.31) it follows that $\mathcal{P}$ as defined

in (2.33) is always a perfect mapping. It is now possible to apply a Quasi-Newton method to the new rootfinding problem

$$\mathcal{K}(\mathbf{x}) = \mathbf{0} \quad \text{with} \quad \mathcal{K}(\mathbf{x}) = \mathcal{P}(\mathbf{x}) - \mathbf{z}^*, \tag{2.34}$$

with $\mathcal{K} : \mathbb{R}^{N_I^S} \rightarrow \mathbb{R}^{N_I^S}$, which is the *outer* method. This results in the Aggressive Space-Mapping (ASM) algorithm as defined in [1, 16]. If $\widetilde{\mathcal{R}}$ is close to $\mathcal{R}$ it holds that

$$\frac{\partial \mathcal{K}}{\partial \mathbf{x}} = \frac{\partial \mathcal{P}}{\partial \mathbf{x}} \approx \mathbf{I}, \tag{2.35}$$

and it is likely that the Quasi-Newton algorithm converges faster when applied to the new rootfinding problem in Eq. (2.34). The ASM algorithm consists of two steps:

1. Solve for the low fidelity fluid-structure interaction solution $\mathbf{z}^*$.
2. Apply a Quasi-Newton algorithm to the new rootfinding problem in Eq. (2.34) .

Although the space mapping function in (2.33) is always a perfect mapping, convergence is not guaranteed. To understand why this is case the concept of model flexibility is used. The definition of model flexibility from [1] states

Definition 2 *A model is called more flexible than another if the set of its reachable aims contains the set of reachable aims of the other. Two models are equally flexible if their sets of reachable aims coincide.*

Hence, a low fidelity model is more flexible than a high fidelity model if $\widetilde{\mathcal{R}}(Z) \supset \mathcal{R}(X)$ and less flexible if $\widetilde{\mathcal{R}}(Z) \subset \mathcal{R}(X)$. From the lemma's in [1] it is also found that

1. If the low fidelity model is more flexible than the high fidelity model then $\mathcal{P} : X \rightarrow Z$ is injective if $\mathcal{R} : X \rightarrow R$ is injective.
2. If the low fidelity model and the high fidelity model are equally flexible and if $\mathcal{R} : X \rightarrow R$ is injective then $\mathcal{P}$ is a bijection.
3. If the high fidelity model is more flexible than the low fidelity model then $\mathcal{P} : X \rightarrow Z$ is surjective.

Convergence problems may for example occur if the space mapping function is surjective in the region of the high fidelity model solution $\mathbf{x}^*$. Perfect mapping is a property that concerns only a *point* while model flexibility concerns a *region*. It can therefore happen that the mapping is perfect since $\mathbf{x}^*$ maps perfectly to $\mathbf{z}^*$ while $\mathbf{x}^*$ is not the only solution that maps to $\mathbf{z}^*$ which means that it is non-unique. It can still be useful to apply the ASM algorithm when this happens if it is combined with conventional algorithms as will become clear in the numerical examples.

To approximate the space-mapping Jacobian $\frac{\partial \mathcal{P}}{\partial \mathbf{x}}$ in step 2 of the ASM algorithm mainly *Broyden's method* is used in the space-mapping community [1, 16]. Although Broyden's method has been used in the FSI community as well [7, 17] and

in the first work on space-mapping accelerated algorithms for FSI [18], the QN-ILS method from section 2.2 is chosen as the outer iterative method in this work.

Broyden's method uses information of only two recent iterates to approximate the Jacobian while the recently developed QN-ILS method uses information from several previous iterates, therefore belonging to the class of multi-iterate methods [19]. This is a heuristic explanation of the reported succes of the QN-ILS method in the FSI community [4, 6, 9] when compared to other Quasi-Newton methods.

To apply the QN-ILS method from section 2.2 to the new rootfinding problem, substitute $\mathbf{x}^{new}$ from Eq. (2.11) into Eq. (2.34) and linearize

$$\mathcal{K}(\mathbf{x}^{new}) = \mathcal{P}(\mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k(\mathbf{x}^i - \mathbf{x}^k)) - \mathbf{z}^* \tag{2.36}$$

$$\approx \mathcal{P}(\mathbf{x}^k) - \mathbf{z}^* + \left(\frac{\partial \mathcal{P}}{\partial \mathbf{x}}\right) \sum_{i=0}^{k-1} c_i^k(\mathbf{x}^i - \mathbf{x}^k)$$

$$\approx \mathbf{p}^k - \mathbf{z}^* + \sum_{i=0}^{k-1} c_i^k(\mathbf{p}^i - \mathbf{p}^k). \tag{2.37}$$

The coefficients c_i^k are subsequently found from minimization of the linearized residual $\mathcal{K}(\mathbf{x}^{new})$

$$\mathbf{c}^k = \arg\min_{\mathbf{c}^k \in \mathbb{R}^k} ||\mathbf{p}^k - \mathbf{z}^* + \sum_{i=0}^{k-1} c_i^k(\mathbf{p}^i - \mathbf{p}^k)||, \tag{2.38}$$

such that $\mathbf{x}^{new}$ can be substituted in Eq. (2.10) and subsequently linearized to find the update

$$\mathbf{x}^{k+1} \approx \mathbf{x}_{\mathcal{H}}^k + \sum_{i=0}^{k-1} c_i^k(\mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k). \tag{2.39}$$

The ASM-ILS algorithm is summarized in Algorithm 4. The numerical evaluation of the space-mapping function on line 2 and line 18 in Algorithm 4 is performed using for example Algorithm 3.

Algorithm 4 ASM - ILS

Require: $\mathbf{x}^0, \mathbf{z}^*, \epsilon_I$
$k = 0$
$\mathbf{p}^k = \mathcal{P}(\mathbf{x}^k)$
while $||\mathbf{r}^k|| > \epsilon_I$ **do**
  if $k = 0$ **then**
    $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{z}^* - \mathbf{p}^k$
  else
    for $i = 0$ to $k - 1$ **do**
      $\Delta\mathbf{x}^i_{\mathcal{H}} = \mathbf{x}^i_{\mathcal{H}} - \mathbf{x}^k_{\mathcal{H}}$
      $\Delta\mathbf{p}^i = \mathbf{p}^i - \mathbf{p}^k$
    end for
    $\mathbf{V}^k = \left[\Delta\mathbf{p}^{k-1}\ \Delta\mathbf{p}^{k-2}\ \ldots\ \Delta\mathbf{p}^0\right]$
    $\mathbf{W}^k = \left[\Delta\mathbf{x}^{k-1}_{\mathcal{H}}\ \Delta\mathbf{x}^{k-2}_{\mathcal{H}}\ \ldots\ \Delta\mathbf{x}^0_{\mathcal{H}}\right]$
    Calculate $\mathbf{V}^k = \mathbf{Q}^k\mathbf{R}^k$
    Calculate $\mathbf{R}^k\mathbf{c}^k = \mathbf{Q}^{kT}(\mathbf{z}^* - \mathbf{p}^k)$
    $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{W}^k\mathbf{c}^k + \mathbf{r}^k$
  end if
  $k = k + 1$
  $\mathbf{p}^k = \mathcal{P}(\mathbf{x}^k)$
end while
return $\mathbf{x}^* = \mathbf{x}^k$

Other inner methods could be used for the numerical evaluation of the space-mapping function such as the QN-ILS method itself, Broyden's method or Aitken's method. As will become clear in chapter 3, the choice for the inner iterative method depends on how expensive the evaluation of the low fidelity fluid operator is. When the cost of the low fidelity fluid operator is very low when compared to the high fidelity fluid operator this choice will not affect the total speedup of the ASM-ILS algorithm significantly as long as the inner method result in a stable algorithm. When the cost of the low fidelity fluid operator is low but not several orders lower than the cost of the high fidelity model it is wise to choose an efficient coupling algorithm for the inner method since it can have a large impact on the total speedup of the ASM-ILS algorithm in this case.

2.3.3. Output Space-Mapping

Sofar, only the *primal* formulation has been used in order to derive space mapping algorithms. A different approach is to define a surrogate model $\mathcal{R}_s^k$ that needs to be updated at every iteration k during the coupling algorithm. In the *dual* space-mapping approach the next iterate is defined as

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} ||\mathcal{R}_s^k(\mathbf{x})||. \tag{2.40}$$

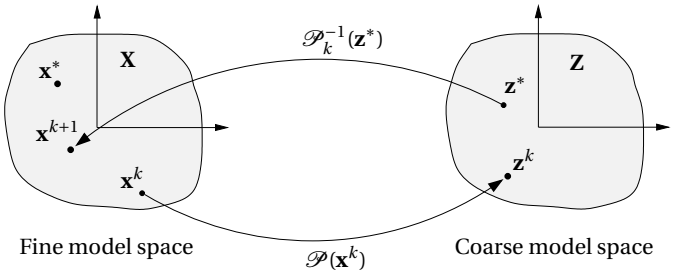


Figure 2.1: Concept of space mapping

For example, substitute $\mathbf{z} = \mathcal{P}_k(\mathbf{x})$ in $\widetilde{\mathcal{R}}(\mathbf{z})$ to obtain $\mathcal{R}_s^k(\mathbf{x}) = \widetilde{\mathcal{R}}(\mathcal{P}_k(\mathbf{x}))$. With

$$\mathcal{P}_k(\mathbf{x}) = \mathcal{P}(\mathbf{x}^k) + \nabla_{\mathbf{x}}^k \mathcal{P}\left(\mathbf{x} - \mathbf{x}^k\right), \tag{2.41}$$

equation (2.40) yields

$$\mathbf{x}^{k+1} = \mathcal{P}_k^{-1}(\mathbf{z}^*) = \mathbf{x}^k + (\nabla_{\mathbf{x}}^k \mathcal{P})^{-1}(\mathbf{z}^* - \mathbf{p}^k), \tag{2.42}$$

which is exactly an Aggressive Space Mapping update. Figure 2.1 shows a schematic of the Aggressive Space Mapping update. The dual and primal formulation yield the same update in this case. However, the dual formulation allows for a different point of view. When the space-mapping function is not perfect or when the minimization in (2.40) is not performed fully until the desired tolerance is reached there is a difference between the output of the surrogate and the output of the fine model: $\mathcal{R}_s^k(\mathbf{x}^k) \neq \mathcal{R}(\mathbf{x}^k)$. In this case one can choose not to correct only for the *input* of the models via a mapping $\mathcal{P}$ but also for the *output* of the models via a mapping $\mathcal{O}$. In this case, the surrogate becomes

$$\mathcal{R}_s^k(\mathbf{x}) = \mathcal{O}^k\left(\widetilde{\mathcal{R}}\left(\mathcal{P}^k(\mathbf{x})\right)\right), \tag{2.43}$$

such that

$$\mathcal{O}^k\left(\widetilde{\mathcal{R}}\left(\mathcal{P}^k(\mathbf{x}^k)\right)\right) = \mathcal{R}(\mathbf{x}^k). \tag{2.44}$$

The output mapping $\mathcal{O}$ can be a linear transformation in a similar way as the input mapping $\mathcal{P}$. For example

$$\mathcal{O}^k(\cdot) = \mathbf{A}^k(\cdot) + \mathbf{d}^k. \tag{2.45}$$

In this way a surrogate is constructed that locally always matches the fine model. More can be found on output space mapping as a coupling method for partitioned Fluid-Structure Interaction in [20]. In this contribution it is shown that both input space mapping and output space mapping are special cases of the generalized space mapping framework as described in [21].

References

[1] D. Echeverría and P. W. Hemker, *Space mapping and defect correction,* Computational Methods in Applied Mathematics **5**, 107 (2005).

[2] J. Donea, A. Huerta, J.-P. Ponthot, and A. Rodriguez-Ferran, *Arbitrary Lagrangian–Eulerian Methods,* in *Encyclopedia of Computational Mechanics, 3-Volume Set*, edited by E. Stein, R. de Borst, and T. J. R. Hughes (Wiley, 2004) Chap. 14.

[3] Y. Bazilevs, J. R. Gohean, T. J. R. Hughes, R. D. Moser, and Y. Zhang, *Patient-specific isogeometric fluid–structure interaction analysis of thoracic aortic blood flow due to implantation of the jarvik 2000 left ventricular assist device,* Computer Methods in Applied Mechanics and Engineering **198**, 3534 (2009).

[4] J. Degroote and J. Vierendeels, *Multi-solver algorithms for the partitioned simulation of fluid–structure interaction,* Computer Methods in Applied Mechanics and Engineering **200**, 2195 (2011).

[5] C. Michler, H. van Brummelen, and R. de Borst, *An investigation of interface-gmres(r) for fluid–structure interaction problems with flutter and divergence,* Computational Mechanics **47**, 17 (2011).

[6] J. Degroote, K.-J. Bathe, and J. Vierendeels, *Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction,* Computers and Structures **87**, 793 (2009), fifth MIT Conference on Computational Fluid and Solid Mechanics.

[7] H. G. Matthies, R. Niekamp, and J. Steindorf, *Algorithms for strong coupling procedures,* Computer Methods in Applied Mechanics and Engineering **195**, 2028 (2006).

[8] R. Haelterman, J. Degroote, D. van Heule, and J. Vierendeels, *The quasi-newton least squares method: A new and fast secant method analyzed for linear systems,* SIAM J. Numer. Anal. **47**, 2347 (2009).

[9] T. Gallinger and K. Bletzinger, *Comparison of algorithms for strongly coupled partitioned fluid-structure interaction,* in *Proceedings of the 5th European Conference on Computational Fluid Dynamics (ECCOMAS CFD 2010)* (2010) 14-17 June 2010, Lisbon, Portugal.

[10] R. Haelterman, J. Degroote, D. van Heule, and J. Vierendeels, *On the similarities between the quasi-newton inverse least squares method and gmres,* SIAM Journal on Numerical Analysis **47**, 4660 (2010).

[11] J. Gerbeau and M. Vidrascu, *A quasi-newton algorithm based on a reduced model for fluid-structure interaction problems in blood flow,* Mathematical Modelling and Numerical Analysis **37**, 631 (2003).

[12] J. E. Dennis and J. J. More, *Quasi-newton methods, motivation and theory,* SIAM Review **19**, pp. 46 (1977).

[13] J. Degroote, A. Souto-Iglesias, W. V. Paepegem, S. Annerel, P. Bruggeman, and J. Vierendeels, *Partitioned simulation of the interaction between an elastic structure and free surface flow,* Computer Methods in Applied Mechanics and Engineering **199**, 2085 (2010).

[14] H. Walker and P. Ni, *Anderson acceleration for fixed-point iterations,* SIAM Journal on Numerical Analysis **49**, 1715 (2011).

[15] X. Cai and D. E. Keyes, *Nonlinearly preconditioned inexact newton algorithms,* SIAM J. Sci. Comput **24**, 183 (2000).

[16] J. W. Bandler, R. M. Biernacki, S. H. Chen, R. H. Hemmers, and K. Madsen, *Electromagnetic optimization exploiting aggressive space mapping,* Microwave Theory and Techniques, IEEE Transactions on **43**, 2874 (1995).

[17] S. Minami and S. Yoshimura, *Performance evaluation of nonlinear algorithms with line-search for partitioned coupling techniques for fluid–structure interactions,* International Journal for Numerical Methods in Fluids **64**, 1129 (2010).

[18] T. P. Scholcz, A. H. Zuijlen, and H. Bijl, *A multi-model incremental adaptive strategy to accelerate partitioned fluid-structure algorithms using space-mapping,* in *Proceedings of the 4th International Conference on Computational Methods for Coupled Problems in Science and Engineering (ECCOMAS 2011)* (2011) 20-22 June 2011, Kos Island, Greece.

[19] M. Bierlaire, F. Crittin, and M. Thémans, *A multi-iterate method to solve systems of nonlinear equations,* European Journal of Operational Research **183**, 20 (2007).

[20] L. Florentie, *Acceleration of partitioned fluid-structure interaction simulations by means of space mapping,* Master thesis (2012).

[21] J. W. Bandler, N. Georgieva, M. A. Ismail, J. E. Rayas-Sanchez, and Q.-J. Zhang, *A generalized space-mapping tableau approach to device modeling,* Microwave Theory and Techniques, IEEE Transactions on **49**, 67 (2001).

3

Space-mapping in FSI: Numerical experiments

In this Chapter the methods developed in Chapter 2 are applied to two academic test cases: The 1-D piston problem and the 2-D panel flutter problem. In order to assess the performance of the space-mapping algorithm we first define a performance metric in Section 3.1. This metric is then used in the numerical experiments in Section 3.2.3 and 3.3.4 for both cases. Finally, conclusions are drawn in Section 3.5.

3.1. Speedup

The speedup of the ASM-ILS method with respect to the QN-ILS method is determined by the decrease of computational effort per time step to obtain the high fidelity model solution $\mathbf{x}^*$ up to a specified tolerance ϵ_I.

Let w_f^i and w_c^j be a measure of the cost (flops or CPU time) necessary to evaluate $\mathcal{R}(\mathbf{x}^i)$ and $\widetilde{\mathcal{R}}(\mathbf{z}^j)$ respectively. The average cost per time step of a high fidelity and low fidelity model residual evaluation is then found from

$$\bar{w}_f = \frac{1}{n_f}\sum_{i=0}^{i=n_f} w_f^i \quad \text{and} \quad \bar{w}_c = \frac{1}{n_c}\sum_{j=0}^{j=n_c} w_c^j, \tag{3.1}$$

where n_f and n_c are the total number of high fidelity (fine) and low fidelity (coarse) iterations respectively (including the iterations necessary to find $\mathbf{z}^*$) per time step. The total cost per time step of the ASM-ILS method is subsequently estimated by

$$W^A \approx \bar{w}_f^A n_f^A + \bar{w}_c^A n_c^A, \tag{3.2}$$

whereas the total cost of the QN-ILS method is estimated by

$$W^Q \approx \bar{w}_f^Q n_f^Q. \tag{3.3}$$

In Eq. (3.2) and Eq. (3.3), the superscript A refers to a quantity associated with the Aggressive Space Mapping algorithm and the superscript Q refers to a quantity associated with a conventional Quasi-Newton algorithm.

The estimates in Eq. (3.2) and Eq. (3.3) are based on the premise that the largest part of the computational effort is spent in order to evaluate the low fidelity and high fidelity residuals in the computation, neglecting all other (overhead) costs. Numerical experiments justify this premise.

The speedup of the ASM-ILS algorithm relative to the QN-ILS algorithm is subsequently found from the ratio of work per time step

$$S_p \approx \frac{W^Q}{W^A} = \frac{n_f^Q}{n_f^A + \frac{\bar{w}_c^A}{\bar{w}_f^Q} n_c^A}, \tag{3.4}$$

which is valid when $\bar{w}_f^A \approx \bar{w}_f^Q$. The ASM-ILS method is more efficient than the QN-ILS method if $S_p > 1$. The speedup becomes insensitive to the number of low fidelity residual evaluations n_c^A if the ratio $\frac{\bar{w}_c^A}{\bar{w}_f^Q}$ is sufficiently small. The choice for the inner iterative method does not affect the speedup in this case as long as it results in a stable algorithm for the evaluation of the space mapping function. The expression of speedup in Eq. (3.4) reflects the principle of space-mapping: If the low fidelity residual function is cheap to evaluate and sufficiently accurate we have $n_f^A < n_f^Q$ and $\frac{\bar{w}_c^A}{\bar{w}_f^Q} n_c^A << 1$ resulting in $S_p > 1$. To obtain a metric of the total speedup of a simulation we define

$$\widehat{S_p} \approx \frac{\sum W^Q}{\sum W^A}, \tag{3.5}$$

where the sum is taken over all the time steps in the numerical simulation. The total estimated speedup $\widehat{S_p}$ is used to compare the performance of the coupling algorithms while the speedup per time step S_p can be printed after completion of each time-step which gives an indication of the success of the space-mapping algorithm during the simulation.

3.2. 1D piston problem

The 1-D piston problem is a test case used in [1] and [2]. A compressible inviscid fluid in a closed section is in interaction with a piston having a certain mass m and stiffness k, see Figure 3.1.

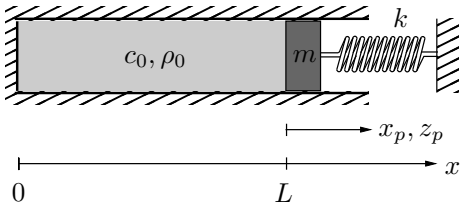


Figure 3.1: The piston problem

The density of the fluid at the equilibrium position of the piston is denoted by ρ_0 and the speed of sound by c_0.

In Figure 3.1, x_p refers to the piston displacement computed with the fine fluid model and z_p refers to the piston displacement computed with the coarse fluid model. The dynamics of the mass-spring system is described by

$$m\ddot{x}_p + kx_p = \Delta p \qquad \text{on } \Gamma_s^I. \tag{3.6}$$

The external pressure difference Δp over the piston is given by

$$\Delta p = p_I - p_0 \quad \text{on } \Gamma_s^I. \tag{3.7}$$

Rewriting to a twice as large system of equations yields

$$\partial_t \mathbf{w}^p + \mathbf{A}^{pp}\mathbf{w}^p = \mathbf{f}_{\text{ex}}^p, \tag{3.8}$$

with $\mathbf{w}^p = [x_p \ \dot{x}_p]^T$ and the external force vector $\mathbf{f}_{\text{ex}}^p = [0 \ \Delta p]^T$. Using the BDF-2 time-discretization scheme, the residual of the discrete structure equations reads

$$\mathbf{r}^p(\mathbf{w}^p; p_I) = \mathbf{w}^p + \frac{\Delta t}{\beta}\left(\mathbf{A}^{pp}\mathbf{w}^p - \mathbf{f}_{\text{ex}}^p\right) - \mathbf{w}^{p,n} - \frac{\Delta t}{\beta}\left(\frac{\mathbf{w}^{p,n} - \mathbf{w}^{p,n-1}}{2\Delta t}\right), \tag{3.9}$$

where $\beta = \frac{3}{2}$. The structure operator $x_p = \mathcal{S}(p_I)$ is subsequently defined by

$$x_p = \xi_x(\mathbf{w}^p) \quad \text{with} \quad \mathbf{w}^p = \arg\min_{\mathbf{w}^p \in \mathbb{R}^2} ||\mathbf{r}^p(\mathbf{w}^p, p_I)||. \tag{3.10}$$

Computation of the interface pressure p_I from the fine fluid model is the topic of Section 3.2.1 and computation of the interface pressure $\tilde{p}_I$ using the coarse fluid model is discussed in Section 3.2.2.

3.2.1. Fine fluid model

The fluid in the piston is governed by the Euler equations of gas dynamics [3, 4]

$$\begin{cases} \partial_t \mathbf{q} + \partial_x \mathbf{f}(\mathbf{q}) = \mathbf{0} & \text{in } \Omega_f, \\ \frac{\partial \rho}{\partial x} = 0 \quad \text{and} \quad u = 0 & \text{on } \Gamma_f^w, \\ \frac{\partial \rho}{\partial x} = 0 \quad \text{and} \quad u = \dot{x}_p & \text{on } \Gamma_f^I. \end{cases} \tag{3.11}$$

Using the isentropic relation $p = \frac{p_0}{\rho_0^\gamma}\rho^\gamma$, the energy equation in Eq. 3.11 becomes redundant and only the mass and momentum balance are necessary to describe the physics of the fluid. The state and flux vector are therefore given by

$$\mathbf{q} = \begin{bmatrix} \rho \\ \rho u \end{bmatrix} \quad \text{and} \quad \mathbf{f}(\mathbf{q}) = \begin{bmatrix} \rho u \\ \rho u^2 + p \end{bmatrix} \tag{3.12}$$

where ρ denotes the fluid density and u the horizontal fluid velocity.

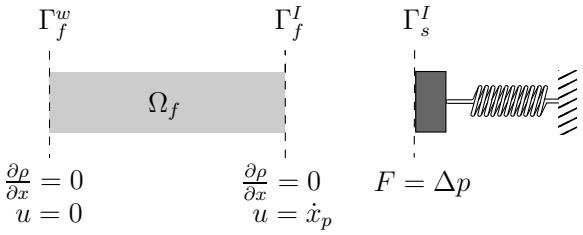


Figure 3.2: The piston problem: boundary conditions

A transpiration boundary condition is used on the fluid-structure interaction interface Γ_f^I. We neglect the term $\partial_t \rho u$ in the momentum equation on Γ_f^I . The transpiration boundary condition implies that we fix the location of Γ_f^I such that $\Gamma_f^I \neq \Gamma_s^I$ and continuity of velocity is satisfied if the piston velocity $\dot{x}_p$ equals the fluid velocity u at Γ_f^I, see Figure 3.2. The fluid boundary value problem (3.11) is discretized using the Finite Volume method. A static mesh with volume width $\Delta x = \frac{L}{N_v}$ is used, where N_v denotes the number of finite volumes.

Integration over a finite volume $\Omega_v^i \subset \Omega_f$ yields

$$\partial_t \int_{\Omega_v^i} \mathbf{q} \mathrm{d}x - \mathbf{f}_{i-1/2} + \mathbf{f}_{i+1/2} = \mathbf{0}. \tag{3.13}$$

Using a second order central discretization, Eq. (3.13) is written as

$$\partial_t \mathbf{q}_i \Delta x + \underbrace{\frac{1}{2}\begin{bmatrix} -\mathbf{J}_{i-1} & \mathbf{0} & \mathbf{J}_{i+1} \end{bmatrix}}_{\Delta x \mathbf{A}_i^{ff}} \begin{bmatrix} \mathbf{q}_{i-1} \\ \mathbf{q}_i \\ \mathbf{q}_{i+1} \end{bmatrix} = \mathbf{0}, \tag{3.14}$$

with

$$\mathbf{J}_i(\mathbf{q}_i) = \begin{bmatrix} 0 & 1 \\ \frac{p_0}{\rho_0^\gamma}\rho_i^{\gamma-1} - u_i^2 & 2u_i \end{bmatrix}, \tag{3.15}$$

Matrix assembly in Eq. (3.14) yields a semi-discrete system of equations

$$\partial_t \mathbf{w}^f + \mathbf{A}^{ff}(\mathbf{w}^f)\mathbf{w}^f = \mathbf{f}_{\text{ex}}^f. \tag{3.16}$$

Here, $\mathbf{w}^f = (\mathbf{q}_1^T\ \mathbf{q}_2^T ... \mathbf{q}_{N_v}^T)^T$ is the discrete state vector of the fluid, $\mathbf{f}_{\text{ex}}^f = \mathbf{f}_{\text{ex}}^f(x_p)$ is the external force vector which depends on the state of the structure. Upon time-discretization of the system of equations in Eq. (3.16), using the BDF-2 time-integration scheme, the residual of the discrete fluid equations is defined by [5]

$$\mathbf{r}^f(\mathbf{w}^f; x^p) = \mathbf{w}^f + \frac{\Delta t}{\beta}\left(\mathbf{A}^{ff}(\mathbf{w}^f)\mathbf{w}^f - \mathbf{f}_{\text{ex}}^f\right) - \mathbf{w}^{f,n} - \frac{\Delta t}{\beta}\left(\frac{\mathbf{w}^{f,n} - \mathbf{w}^{f,n-1}}{2\Delta t}\right) \tag{3.17}$$

The fine fluid-operator $p_I = \mathcal{F}(x_p)$ is defined by

$$p_I = \xi_p(\mathbf{w}^f) \quad \text{with} \quad \mathbf{w}^f = \arg\min_{\mathbf{w}^f \in \mathbb{R}^{2N_v}} ||\mathbf{r}^f(\mathbf{w}^f; x^p)||, \tag{3.18}$$

where the map $p_I = \xi_p(\mathbf{w}^f)$ is given by

$$p_I = \frac{p_0}{\rho_0^\gamma}\rho_I^\gamma \quad \text{with} \quad \rho_I = (0, ..., 0\ 1\ 0) \cdot \mathbf{w}^f. \tag{3.19}$$

The density ρ_I is the density of the fluid ρ_{N_v} near the fluid-structure interaction interface. The minimization in Eq. (3.18) is performed using Picard iterations. With the definition of the structure operator in Section Eq. (3.10) and the fine fluid operator in Eq. (3.18) , the fine interface residual function is defined as

$$\mathcal{R}(x_p) = \mathcal{S} \circ \mathcal{F}(x_p) - x_p, \tag{3.20}$$

which is used in the evaluation of the space-mapping function in algorithm 3 and the ASM-Secant algorithm 4.

3.2.2. Coarse fluid model

The coarse fluid model is found from a linearization of the fluid with respect to the equilibrium state of the piston. If the flux vector $\mathbf{f}$ in Eq. (3.11) is linearized around the equilibrium state of the fluid: $\mathbf{q}_0 = (\rho_0\ 0)^T$ we obtain

$$\partial_t \mathbf{q}' + \partial_{\mathbf{q}}\mathbf{f}|_{\mathbf{q}=\mathbf{q}_0}\ \partial_x \mathbf{q}' = \mathbf{0}. \tag{3.21}$$

The Jacobian $\partial_{\mathbf{q}}\mathbf{f}$ of the nonlinear flux $\mathbf{f}$ is given by [3, 4]

$$\partial_{\mathbf{q}}\mathbf{f}|_{\mathbf{q}=\mathbf{q}_0} = \begin{bmatrix} 0 & 1 \\ c_0^2 & 0 \end{bmatrix}, \tag{3.22}$$

and $c_0 = \sqrt{\frac{\gamma p^0}{\rho^0}}$ and $\mathbf{q}'$ a perturbation with respect to the equilibrium state vector $\mathbf{q}_0$.

$$\mathbf{q} = \mathbf{q}_0 + \mathbf{q}'. \tag{3.23}$$

Integrating Eq. (3.21) over a finite volume and using a central discretization this can be written

$$\dot{\mathbf{q}}_i' \Delta x + \underbrace{\frac{1}{2}\begin{bmatrix} -\partial_{\mathbf{q}}\mathbf{f} & \mathbf{0} & \partial_{\mathbf{q}}\mathbf{f} \end{bmatrix}}_{\Delta x \widetilde{\mathbf{A}}_i^{ff}} \begin{bmatrix} \mathbf{q}_{i-1}' \\ \mathbf{q}_i' \\ \mathbf{q}_{i+1}' \end{bmatrix} = \mathbf{0}, \tag{3.24}$$

resulting in a linear semi-discrete system of equations after matrix assembly

$$\partial_t \widetilde{\mathbf{w}}^f + \widetilde{\mathbf{A}}^{ff} \widetilde{\mathbf{w}}^f = \widetilde{\mathbf{f}}^f_{\text{ex}}. \tag{3.25}$$

Here, the state vector is denoted $\widetilde{\mathbf{w}}^p = [z^p \ \dot{z}^p]^T$ and $\widetilde{\mathbf{f}}^f_{\text{ex}} = \widetilde{\mathbf{f}}^f_{\text{ex}}(z_p)$. The residual of the discrete fluid equations is defined by

$$\widetilde{\mathbf{r}}^f(\widetilde{\mathbf{w}}^f; z^p) = \left(\mathbf{I} + \frac{\Delta t}{\beta}\widetilde{\mathbf{A}}^{ff}\right)\widetilde{\mathbf{w}}^f - \frac{\Delta t}{\beta}\widetilde{\mathbf{f}}^f_{\text{ex}} - \widetilde{\mathbf{w}}^{f,n} - \frac{\Delta t}{\beta}\left(\frac{\widetilde{\mathbf{w}}^{f,n} - \widetilde{\mathbf{w}}^{f,n-1}}{2\Delta t}\right). \tag{3.26}$$

The coarse fluid-operator $\widetilde{p}_I = \widetilde{\mathcal{F}}(z_p)$ is now defined by

$$\widetilde{p}_I = \widetilde{\xi}_p(\widetilde{\mathbf{w}}^f) \quad \text{with} \quad \widetilde{\mathbf{w}}^f = \arg\min_{\widetilde{\mathbf{w}}^f \in \mathbb{R}^{2N_v}} ||\widetilde{\mathbf{r}}^f(\widetilde{\mathbf{w}}^f; z^p)||, \tag{3.27}$$

where the map $\widetilde{p} = \widetilde{\xi}_p(\widetilde{\mathbf{w}}^f)$ is now given by

$$\widetilde{p}_I = p_0 + c_0^2 \rho_I' \quad \text{with} \quad \rho_I' = (0, ..., 0 \ 1 \ 0) \cdot \widetilde{\mathbf{w}}^f. \tag{3.28}$$

The minimization in Eq. (3.27) only requires a linear system solve. The coarse interface residual function is now defined as

$$\widetilde{\mathcal{R}}(z_p) = \mathcal{S} \circ \widetilde{\mathcal{F}}(z_p) - z_p, \tag{3.29}$$

which is used to compute z^* and to evaluate the space-mapping function in algorithm 3.

3.2.3. Numerical experiments

The fluid-to-structure mass ratio ζ and ratio of characteristic time-scales λ are defined by

$$\zeta = \rho_0 L/m \quad \text{and} \quad \lambda = L\omega/c_0 \tag{3.30}$$

with $\omega = \sqrt{\frac{k}{m}}$ the natural frequency of the mass-spring system and $\lambda = 2\pi\tau_f/\tau_s$ with $\tau_f = L/c_0$ the time for a pressure wave to travel from one side of the fluid domain to the other and $\tau_s = 2\pi/\omega$ the natural period of the mass spring system. It is well known that the convergence of fixed-point iterations depends on the mass ratio ζ, the time ratio λ and the time step used in the sequential integration process, see [2, 3]. For sufficiently small time steps only two fixed point iterations are necessary to reach convergence. In order to study the performance of the ASM-algorithm we use a range of time steps sizes and fluid-structure interaction coupling strengths. The coupling strength is controlled by fixing the ratio of characteristic time scales and increasing the fluid-to-structure mass ratio, see table 3.1.

Test case	ζ	λ
FSI-weak	0.50	0.85
FSI-medium	0.67	0.85
FSI-strong	2.00	0.85

Table 3.1: Similarity parameters of the 1-D FSI test cases

In the following the nondimensional variables

$$\bar{t} = \frac{t}{\tau_f}, \quad \bar{x}_p = \frac{x_p}{L}, \quad \bar{v}_p = \frac{\dot{x}_p}{c_0} \quad \bar{\rho} = \frac{\rho}{\rho_0} \tag{3.31}$$

are used and we omit the bars in Eq. (3.31). Time steps in the simulation are given with respect to the nondimensional coupled period of the coarse fluid model $P = T_p/\tau_f = 2\pi/\omega_c$, where T_p denotes the dimensional coupled period and ω_c satisfies [1, 5]

$$\left(1 - \frac{\lambda^2}{\omega_c^2}\right) \omega_c \tan \omega_c = \zeta. \tag{3.32}$$

This results in a coupled period of $P_1 = 6.19$, $P_2 = 5.96$ and $P_3 = 5.04$ for the test cases FSI-weak, FSI-medium and FSI-strong, respectively. The numerical parameters and nondimensional intitial conditions are collected in table 3.2.

For each test case we are interested in the relative time-dependent efficiency η and the total relative efficiency $\hat{\eta}$ of a simulation as defined in Section 3.1.

Description	Symbol	Value
Initial piston displacement	$x_p(t^0)$	0.5
Initial piston velocity	$v_p(t^0)$	0
Initial fluid density	$\rho(x, t^0)$	$\frac{2}{3}$
Finite volume cells	N_v	64
Number of time steps	N_t	$10 \cup 30 \cup 70$
Time step	Δt	P/N_t
Outer tolerance	ϵ_I	$1 \cdot 10^{-6}$
Inner tolerance	ϵ_s	$1 \cdot 10^{-7}$

Table 3.2: Initial conditions and numerical parameters.

Figure 3.3 shows the piston displacement x_p and velocity v_p calculated with the fine fluid model for $t \in [0\ 2P]$. The density fields calculated with the fine and coarse fluid model are given in Figure 3.4 and 3.5 respectively.

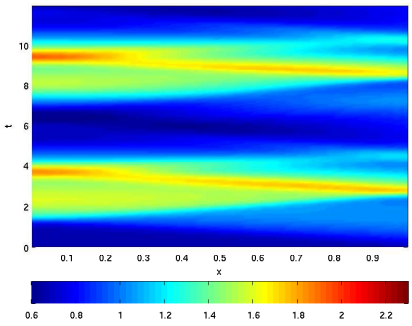


Figure 3.4: Fluid density $\rho(x,t)$ computed with the fine fluid model, FSI-medium, $\Delta t = P/70$.

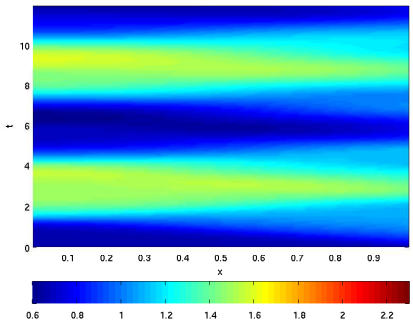


Figure 3.5: Fluid density $\rho(x,t)$ computed with the coarse fluid model, FSI-medium, $\Delta t = P/70$.

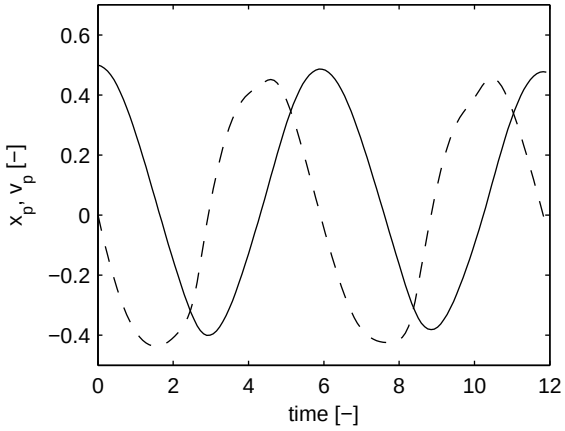


Figure 3.3: Nondimensional piston displacement x_p $(-)$ and piston velocity $v_p = \frac{\partial x_p}{\partial t}$ $(--)$, FSI-medium, $\Delta t = P/70$.

Comparing Figure 3.4 with Figure 3.5 it can be seen that the coarse fluid model does not capture the pressure wave accurately due to the linearized flux in Eq. (3.21).

In the following we will compare the estimated speedup from Section 3.1 to the observed speedup for the FSI test cases defined in table 3.1. The estimated speedup of the test cases are assembled in table 3.3. The speedup obtained with the ASM-Secant method is relatively high for weakly coupled problems at large time steps and for strongly coupled problems. For weakly coupled problems (FSI-weak) at small time steps ($P/70$) it becomes difficult to gain efficiency with respect to the Quasi-Newton method. Both methods show similar convergence to the fixed point iteration method from algorithm 1 in this case. When large time steps ($P/10$) are used the initial guess of the coupled problem becomes worse which results in

a need for more coupling iterations before converge is achieved. The speedup is observed to be higher in these cases. For strongly coupled problems (FSI-strong) at large time steps ($P/10$) the Secant method becomes unstable while the ASM-Secant converges. This indicates that the ASM-Secant method has better robustness properties than the Secant method.

The speedup of the ASM-Secant method with respect to the Secant method is at most 1.16, which is not a large improvement. This can be explained by the fact that the 1-D piston problem has only one interface degree of freedom ($N_I = 1$) which means that the Jacobian $\frac{\partial \mathcal{R}}{\partial x_p}$ is a scalar. In practical applications the fluid-structure interaction interface is a surface and we have $N_I > 1$ resulting in a Jacobian represented by a matrix of size $N_I \times N_I$. Therefore, a more representative 2-D test case is considered in Section 3.3.

Test case	$P/10$	$P/30$	$P/70$
FSI-weak	1.16	1.03	1.02
FSI-medium	1.10	1.11	1.01
FSI-strong	1.04*	1.13	1.07

Table 3.3: Estimated total speedup $\widehat{S}_p$. *) The secant method becomes unstable in the first iteration while the ASM-Secant method converges. The initial Jacobian in the secant method is therefore changed to $\frac{\partial \mathcal{R}}{\partial x_p}^0 = -0.5$ in order to be able to compute the estimated speedup.

The observed total speedup – the ratio of total CPU times – is assembled in table 3.4. The values listed in table 3.4 vary from simulation to simulation and are therefore not a reliable metric for comparsion. In most cases, however, the values in table 3.4 do not deviate too much from the values listed in table 3.3.

Test case	$P/10$	$P/30$	$P/70$
FSI-weak	1.19	1.05	1.01
FSI-medium	1.13	1.09	1.01
FSI-strong	1.03*	1.28	1.09

Table 3.4: Observed total speedup $\widehat{S}_p$ computed using total CPU times. *) The secant method becomes unstable in the first iteration while the ASM-Secant method converges. The initial Jacobian in the secant method is therefore changed to $\frac{\partial \mathcal{R}}{\partial x_p}^0 = -0.5$ in order to be able to compute the estimated speedup.

Figure 3.6 to 3.8 show the number of fine and coarse fluid operator evaluations for the Gauss-Seidel, Secant and ASM-Secant algorithm applied to the FSI-medium test case at various time step sizes. It can clearly be seen that the ASM-Secant method requires the least number of fine fluid operator evaluations followed by the Secant method and the Gauss-Seidel algorithm for all time step sizes considered.

From Figure 3.6 it is found that both the ASM-Secant and the Secant method perform much better than the Gauss-Seidel algorithm which has a slow convergence

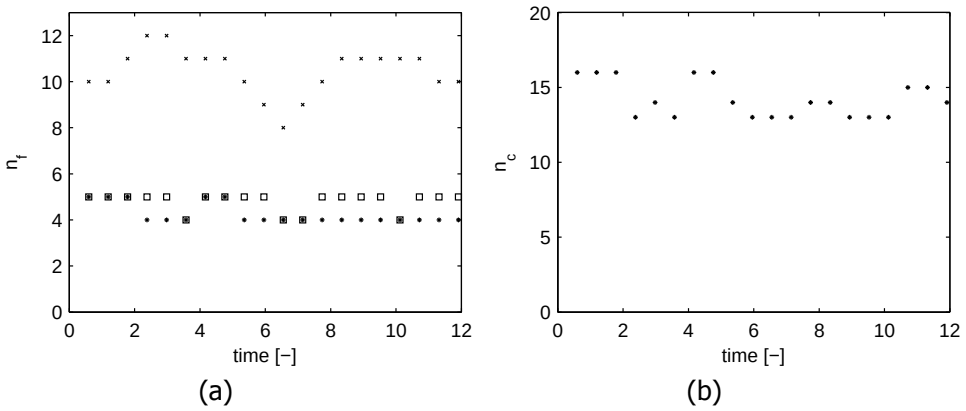


Figure 3.6: (a) Number of fine fluid operator evaluations n_f of the Gauss-Seidel algorithm ($\times$), Secant algorithm ($\square$) and the ASM-Secant algorithm ($*$). Testcase FSI-medium, $\Delta t = P/10$. (b) Number of coarse fluid operator evaluations n_c of the ASM-Secant algorithm.

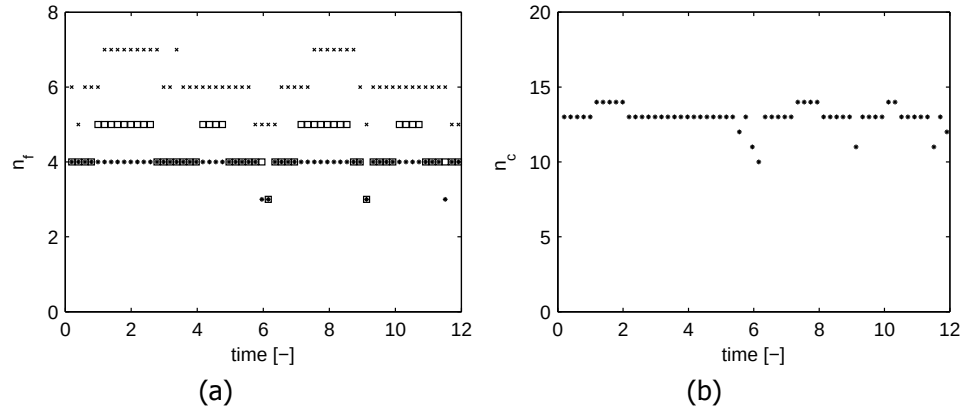


Figure 3.7: (a) Number of fine fluid operator evaluations n_f of the Gauss-Seidel algorithm ($\times$), Secant algorithm ($\square$) and the ASM-Secant algorithm ($*$). Testcase FSI-medium, $\Delta t = P/30$. (b) Number of coarse fluid operator evaluations n_c of the ASM-Secant algorithm.

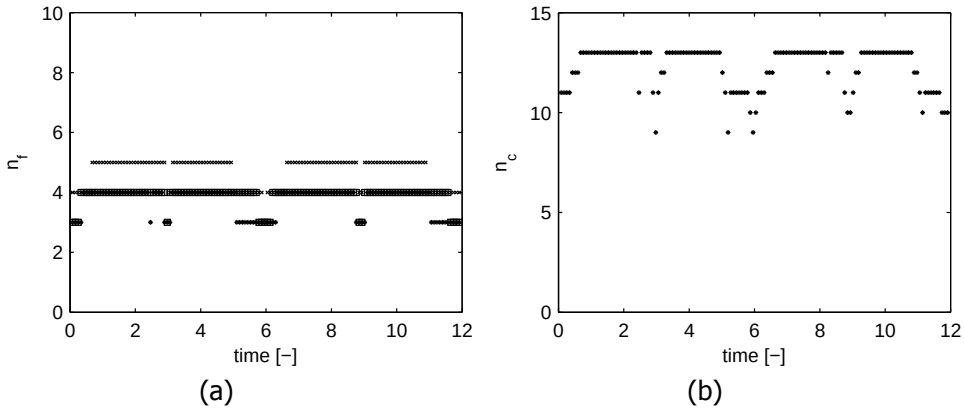


Figure 3.8: (a) Number of fine fluid operator evaluations n_f of the Gauss-Seidel algorithm ($\times$), Secant algorithm ($\square$) and the ASM-Secant algorithm ($*$). Testcase FSI-medium, $\Delta t = P/70$. (b) Number of coarse fluid operator evaluations n_c of the ASM-Secant algorithm.

for large time steps. Reduction of the time step results in a similar amount of fine fluid-operator evaluations for each method as can be seen in Figure 3.8. The ASM-Secant method still performs slightly better in this case.

The number of coarse fluid operator evaluations is much higher than the number of fine fluid operator evaluations for all time step sizes considered. This can be explained by the fact that coarse fluid operator evaluations are necessary to evaluate the space-mapping function at each iteration of the coupling algorithm and to solve for the coarse model solution z^*. The negative effect on the speedup is limited since coarse fluid operator evaluations require far less computational resources than fine fluid operator evaluations.

Figure 3.9 to 3.11 show the L^2 norm of the interface residual as a function of the number of coupling iterations at a representative time step of the simulation. Several time step sizes are studied. The convergence trend of the Secant method is always inbetween the convergence trend of the Gauss-Seidel method and the ASM-Secant method, as expected. The slopes of the convergence trends become more similar to the slope of the Gauss-Seidel method as the time step decreases. This is typical for compressible fluid-structure interaction problems. The Gauss-Seidel method is appropriate for time-step sizes that are sufficiently small.

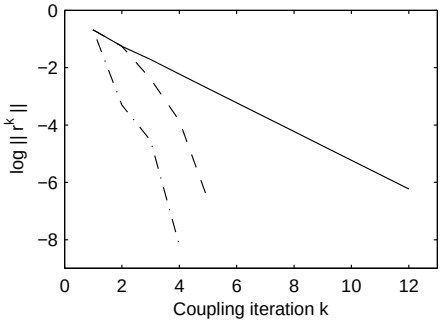


Figure 3.9: Interface residual convergence of a representative time step of test case FSI-medium, $\Delta t = P/10$. Gauss-Seidel algorithm (−), Secant method (−−), ASM-Secant algorithm (− · −).

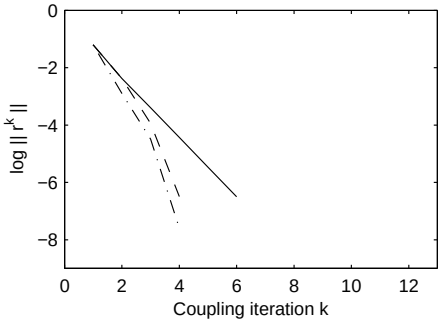


Figure 3.10: Interface residual convergence of a representative time step of test case FSI-medium, $\Delta t = P/30$. Gauss-Seidel algorithm (−), Secant method (−−), ASM-Secant algorithm (− · −).

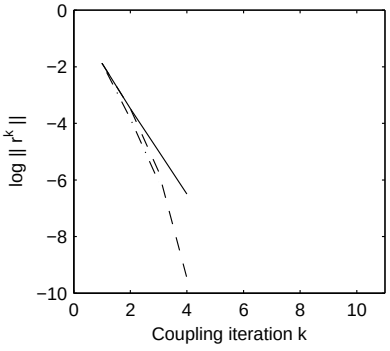


Figure 3.11: Interface residual convergence of a representative time step of test case FSI-medium, $\Delta t = P/70$. Gauss-Seidel algorithm (−), Secant method (−−), ASM-Secant algorithm (− · −).

3.3. 2D panel flutter problem

In this Section the ASM-ILS algorithm is applied to a simple academic test problem – the supersonic panel flutter problem – in order to assess the speedup as defined in Section 3.1. To this end, a structure model, high fidelity fluid model and low fidelity fluid model are defined in Section 3.3.1, 3.3.2 and 3.3.3 respectively. Finally, numerical experiments are performed in Section 3.3.4 in order to investigate the influence of physical parameters and time step sizes on the speedup of the ASM-ILS algorithm.

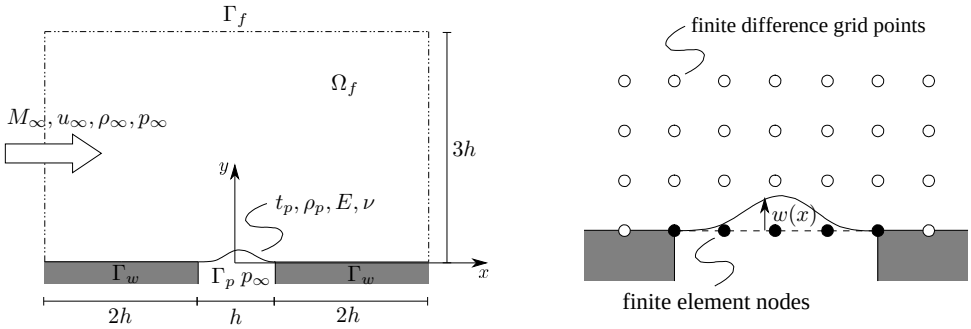


Figure 3.12: Schematic representation of the panel flutter problem

3.3.1. Structure model

The flexible panel is governed by the Euler-Bernoulli beam equation

$$\begin{cases} \rho_p t_p \frac{\partial^2 w}{\partial t^2} + \frac{\partial^2}{\partial x^2}\left[\frac{EI}{1-\nu^2}\frac{\partial^2 w}{\partial x^2}\right] = -\Delta p(x) \quad \text{on } \Gamma_p, \\ w(\pm\frac{h}{2}) = \frac{\partial w}{\partial x}(\pm\frac{h}{2}) = 0. \end{cases} \tag{3.33}$$

In Eq. (3.33), ρ_p denotes the density of the panel, t_p the thickness of the panel, E the Youngs modulus, ν Poisson's ratio, $I = \frac{1}{12}t_p^3$ the moment of inertia and w the vertical panel displacement. The forcing term is given by the pressure difference over the panel

$$\Delta p = p_I(x) - p_\infty \quad \text{on } \Gamma_p, \tag{3.34}$$

where p_I denotes the pressure of the fluid on the fluid-structure interface Γ_p and p_∞ the freestream pressure, see Figure 3.12. The boundary value problem in Eq. (3.33) is discretized using the finite element method with Hermitian shape functions such that the nodal unknowns are displacements *and* rotations.

3.3.2. High fidelity fluid model

The high fidelity fluid is governed by the two-dimensional unsteady linearized potential equation

$$\begin{cases} \nabla^2\phi - \frac{1}{a_\infty^2}\frac{D}{Dt}\left(\frac{D\phi}{Dt}\right) = 0 & \text{in } \Omega_f, \\ \phi(x,y) = 0 & \text{on } \Gamma_f, \\ v(x,y) = 0 & \text{on } \Gamma_w, \\ v(x,y) = \frac{Dw}{Dt} & \text{on } \Gamma_p, \end{cases} \tag{3.35}$$

where ϕ denotes the perturbed fluid potential and M_∞ and a_∞ are the freestream Mach number and freestream fluid speed of sound. The substantial derivative in Eq. (5.4) is given by

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + M_\infty a_\infty \frac{\partial}{\partial x}. \tag{3.36}$$

The linearized potential equation is valid for subsonic flow and supersonic flow but is not applicable in the transonic regime. The horizontal velocity compononent u and vertical component v of the fluid are recovered from the potential according to

$$u = u_\infty + \frac{\partial\phi}{\partial x} \quad \text{and} \quad v = \frac{\partial\phi}{\partial y}. \tag{3.37}$$

The interface pressure on top of the panel is given by Bernoulli's equation [11]

$$p_I(x) = p_\infty - \rho_\infty \frac{D\phi}{Dt} \qquad \text{on } \Gamma_p. \tag{3.38}$$

The fluid boundary value problem in Eq. (5.4) is discretized using the Finite Difference method. The finite difference grid conforms with the finite element mesh of the panel.

3.3.3. Low fidelity fluid model

Using the piston analogy model (see [6]), the interface pressure is approximated by

$$p_I(x) = p_\infty + \rho_\infty M_\infty a_\infty \left(\frac{M_\infty^2 - 2}{\sqrt{(M_\infty^2 - 1)^3}} \frac{\partial w}{\partial t} + M_\infty a_\infty \frac{\partial w}{\partial x} \right), \tag{3.39}$$

with (x) on Γ_p. The piston analogy is valid for $M_\infty > 1.6$. The pressure in Eq. (3.39) directly depends on the deflection of the panel. The computational effort is therefore negligible compared to the use of the high fidelity fluid model.

3.3.4. Numerical experiments

The similarity parameters are the Mach number M_∞, the fluid-to-structure mass ratio ζ and the ratio of characteristic time-scales λ defined by

$$\zeta = \frac{\rho_\infty L}{\rho_p t_p} \quad \text{and} \quad \lambda = \frac{L a_\infty^{-1}}{(\rho_p t_p)^{1/2} L^2 (EI)^{-1/2}}. \tag{3.40}$$

The values of these parameters – for each test case under consideration – are collected in table 3.5. Linear stability analysis is used to obtain the critical Mach number, i.e. the Mach number which separates the stable from the unstable regime. The critical Mach number $M_{\text{cr}} = 2.27$ and circular frequency $\omega_{cr} = 460\ rad/s$ of test case FSI-weak agree with the values reported in [6].

Test case	M_{cr}	ζ	λ
FSI-weak	2.27	$5.47e^{-2}$	$1.47e^{-2}$
FSI-medium	2.28	$7.41e^{-2}$	$1.47e^{-2}$
FSI-strong	2.33	$3.00e^{-1}$	$1.47e^{-2}$

Table 3.5: Similarity parameters of the 2-D FSI test cases

The Newmark-β time integration scheme is used to integrate the structure and fluid equations of motion. This scheme is known to be second order accurate and unconditionally stable for $\gamma = 1/2$ and $\beta = 1/4$. We use the a-form implementation, see [7].

The panel is released from an initial displacement equal to the flutter mode $w^0 = 0.1\xi$, see Figure 3.13. The corresponding steady fluid potential ϕ is depicted in Figure 3.14.

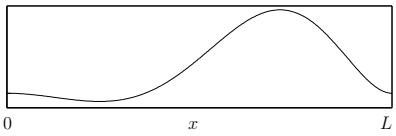


Figure 3.13: Initial panel deflection $w^0 = 0.1\xi$.

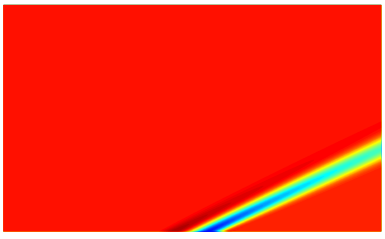


Figure 3.14: Steady initial fluid field.

The simulations are performed at the critical Mach numbers M_{cr}. The observed frequency ω_{cr} in the simulation is then equal to the critical frequency obtained by linear stability analysis. The nondimensional coupled period $P_{cr} = 2\pi a_\infty/\omega_{cr}L$ is divided into 10, 30 and 70 time steps for each test case to study the influence of the time step on the speed up. To study the effect of the fluid grid size on the speedup we do all computations on a small, medium and large fluid grid. The coupled periods of test case FSI-weak, FSI-medium and FSI-strong are $P_{cr}^1 = 8.2$, $P_{cr}^2 = 7.0$ and $P_{cr}^3 = 3.5$ respectively. The numerical parameters used in the simulations are assembled in table 3.6. The medium fluid grid size $N_x \times N_y = 321 \times 193$ and number of finite elements $N_e = 64$ in Table 3.6 correspond to a test case used in [8].

Description	symbol	value
Fluid grid size: small	$N_x \times N_y$	161×97
Fluid grid size: medium	$N_x \times N_y$	321×193
Fluid grid size: large	$N_x \times N_y$	641×385
Number of Finite Elements	N_e	$32 \cup 64 \cup 128$
Number of time steps	N_t	$20 \cup 60 \cup 140$
Time step	Δt	$2P_{cr}/N_t$
Outer tolerance	ϵ_l	$1 \cdot 10^{-6}$
Inner tolerance	ϵ_s	$1 \cdot 10^{-7}$

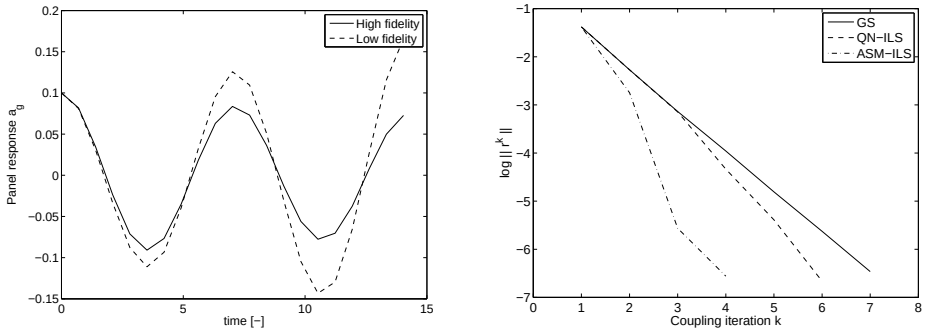
Table 3.6: Numerical parameters

An inner product of the panel displacement with the flutter mode is defined by

$$a_g(t) = \frac{1}{\int_0^L \xi^2(x)dx} \int_0^L \xi(x)w(x,t)dx. \tag{3.41}$$

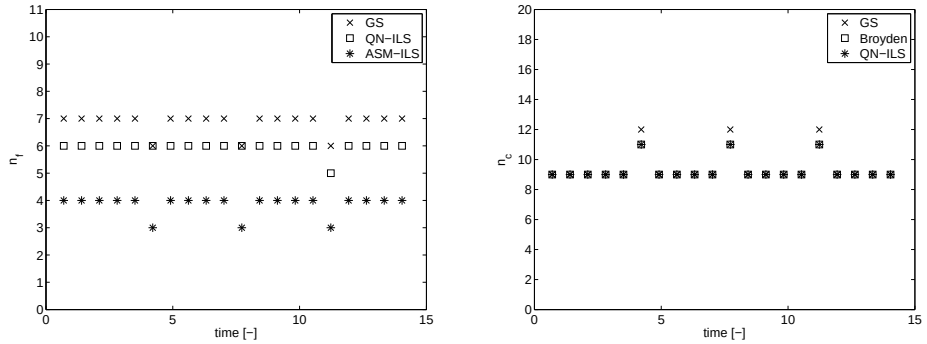
The inner product is used to plot the time history of the panel after it is released from its initial deflection. An example is given in Figure 3.15a which shows the high fidelity and low fidelity model responses. Both responses demonstrate that the panel oscillates in the flutter mode at the predicted frequency $\omega_{cr} = \frac{2\pi a_\infty}{LP_{cr}} = 539 rad/s$.

Figure 3.15b shows the interface residual convergence of a representative time step. It can be seen that the ASM-ILS algorithm outperforms the QN-ILS algorithm. An improvement is already visible after the first iteration and becomes better with the adaption of the space mapping Jacobian. The number of high fidelity fluid operator evaluations as a function of time are plotted in figure 3.16a for the Gauss-Seidel, QN-ILS and the ASM-ILS algorithm. The number of low fidelity fluid operator evaluations used in the ASM-ILS algorithm are plotted in figure 3.16b. Several inner iterative algorithms are investigated: the Gauss-Seidel, Broyden's and the QN-ILS algorithm. The number of low fidelity fluid operator evaluations does not depend too much on the type of inner iterative algorithm. This can be explained by the fact that only two or three low fidelity fluid operator evaluations are required per evaluation of the space-mapping function. In that case, Broyden's algorithm and



(a) (b)

Figure 3.15: (a) Panel response a_g for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Interface residual convergence during a representative time step of test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$



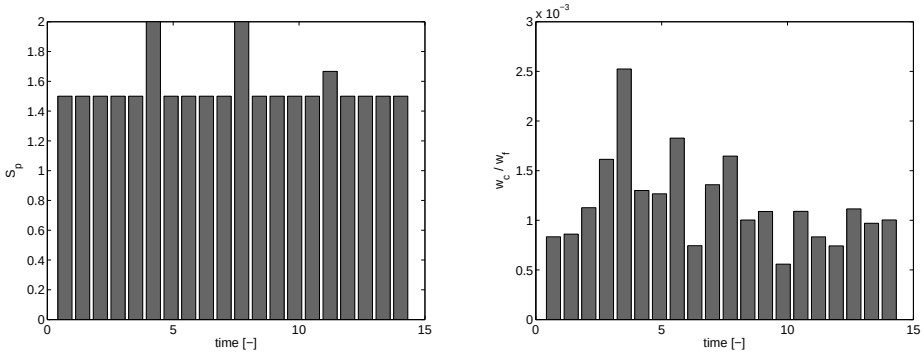
(a) (b)

Figure 3.16: (a) Number of high fidelity fluid operator evaluations for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Number of low fidelity fluid operator evaluations in the ASM-ILS algorithm for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.

the QN-ILS algorithm converge at a similar rate.

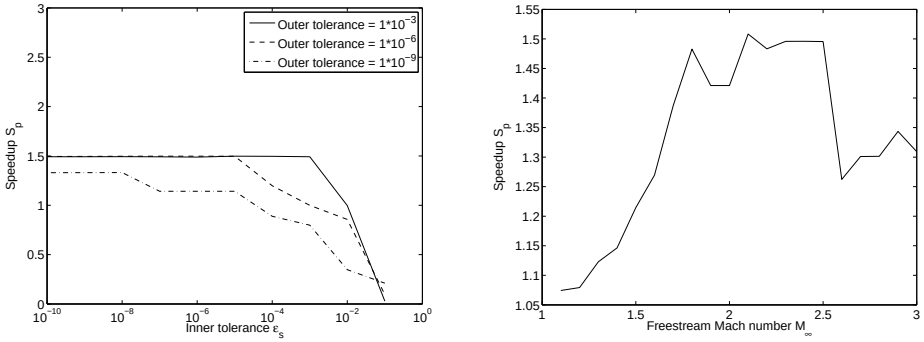
The estimated speedup per time step is shown in Figure 3.17a. The speedup is around $S_p \approx \frac{n_f^Q}{n_f^A} = \frac{6}{4}$ for most time steps in the simulation since $\frac{w_c}{w_f}$ is negligible, see Figure 3.17b.

Table 3.7 lists the total speedup of the simulations for all considered time step sizes, grid sizes and FSI cases. The influence of the time step size is large and demonstrates that no speedup is obtained when the time step size becomes too small. In this case only three iterations are sufficient to converge and it becomes difficult to obtain a speedup larger than 1. In contrast to fluid-structure interaction problems involving incompressible fluids the added mass effect is not causing problems when small time steps are considered, see [9, 10]. The speedup obtained with



(a) (b)

Figure 3.17: (a) Estimated speedup per time step for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Average work ratio per time step for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.



(a) (b)

Figure 3.18: (a) Total speedup of the ASM-ILS algorithm versus the inner tolerance for several outer tolerances using test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Total speedup of the ASM-ILS algorithm versus the freestream Mach number for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.

the ASM-ILS method is therefore limited in this case. The use of a smaller fluid grid results in a smaller speedup when large time step sizes are used in the simulation. This can be attributed to a better numerical conditioning of the FSI problem in this case. There is however almost no difference in speedup between the use of the medium and the large fluid grid, indicating that it is matter of convergence rather than numerical conditioning. The influence of physical parameters on the speedup achieved with the ASM-ILS algorithm is surprisingly small. This requires further investigations.

Grid	FSI	$\Delta t = P_{cr}/10$	$\Delta t = P_{cr}/30$	$\Delta t = P_{cr}/70$
small	weak	1.26	1.30	0.991
	medium	1.27	1.29	0.991
	strong	1.26	1.18	0.992
medium	weak	1.56	1.30	0.997
	medium	1.54	1.31	0.998
	strong	1.44	1.30	0.999
large	weak	1.54	1.30	1.000
	medium	1.51	1.31	1.000
	strong	1.46	1.31	1.000

Table 3.7: Estimated total speedup $\widehat{S_p}$ of the simulations .

Table 3.8 shows the observed speedup measured by the total CPU time ratio for test case FSI-medium on the medium fluid grid. Slightly lower efficiencies are measured due to the overhead costs which are higher for the ASM-ILS algorithm than for the QN-ILS algorithm.

Grid	FSI	$\Delta t = P_{cr}/10$	$\Delta t = P_{cr}/30$	$\Delta t = P_{cr}/70$
medium	weak	1.44	1.23	0.94
	medium	1.45	1.19	0.95
	strong	1.29	1.19	0.97

Table 3.8: Observed speedup computed using total CPU times .

The evaluation of the space-mapping function converges to the desired inner tolerance $\epsilon_s = 1 \cdot 10^{-7}$ in all cases. The low fidelity model is therefore more flexible or equally flexible when compared to the high fidelity model in the region of interest. The space-mapping function is therefore either injective or bijective and the ASM algorithm converges to the high fidelity solution. To study the sensitivity of the speedup with respect to the inner tolerance we plot the speedup versus the inner tolerance for a number of outer tolerances, see Figure 3.18a. From the figure it becomes clear that $\epsilon_s \leq \epsilon_I$ is a safe choice for the inner tolerance. Choosing a larger inner tolerance has a direct negative consequence on the speedup of the simulation. The speedup is plotted versus the freestream Mach number M_∞ in Figure 3.18b.

The speedup increases with the freestream Mach number since the low fidelity fluid model becomes more accurate for $M_\infty > 1.6$ while the computational cost of the model remains constant. However for $M_\infty > M_{\text{cr}}$, the low fidelity and high fidelity model responses are not similar anymore since the low fidelity model starts to flutter at a slight different freestream Mach number than the critical speed M_{cr} of the high fidelity model. This results in a decrease of the speedup. However, the speedup is found to be larger than one for all Mach numbers considered, indicating that the ASM algorithm is robust.

3.4. 1D flexible tube problem

In this section we apply the space-mapping algorithms on the fluid-structure interaction between an incompressible flow in a flexible tube and a massless elastic tube wall. This test case is described in detail in [11, 12]. Figure 3.19 shows the conceptual and computational domain of the flexible tube problem.

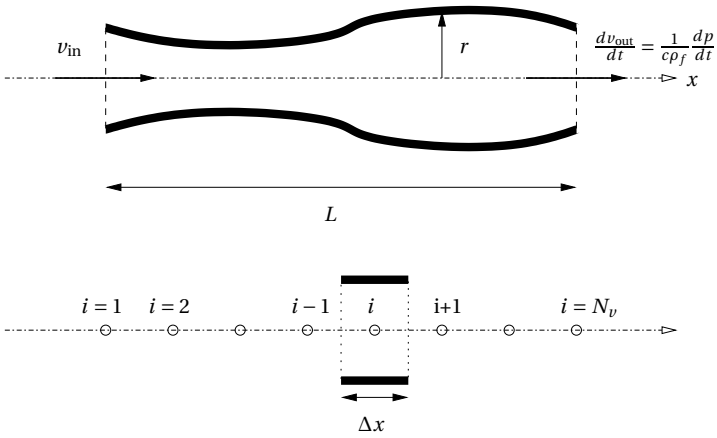


Figure 3.19: The conceptual (top) and computational domain (bottom) of the 1D flexible tube problem.

3.4.1. Structure model

The cross-sectional area $a = \pi r^2$ and the pressure p in the tube are related by

$$a = a_0 \left(\frac{\frac{p_0}{2\rho_f} - c_{MK}^2}{\frac{p}{2\rho_f} - c_{MK}^2} \right)^2, \tag{3.42}$$

where ρ_f is the fluid density and p_0 and a_0 are the initial pressure and initial cross-sectional area respectively. The Moens-Korteweg wave speed is given by

$$c_{MK} = \sqrt{\frac{Eh}{2\rho_f r_0}}, \tag{3.43}$$

where E denotes the Young's modulus and h and r_0 are the wall thickness and initial radius of the tube.

3.4.2. Fluid model

The fluid is considered an incompressible Newtonian fluid. Conservation of mass and momentum yield

$$\frac{\partial a}{\partial t} + \frac{\partial av}{\partial x} = 0, \tag{3.44}$$

$$\frac{\partial av}{\partial t} + \frac{\partial av^2}{\partial x} + \frac{1}{\rho_f}\left(\frac{\partial ap}{\partial x} - p\frac{\partial a}{\partial x}\right) = 0, \tag{3.45}$$

which are the continuity equation and the Navier-Stokes equation in conservative form. Here, x denotes the coordinate along the axis of the tube. The velocity along the axis of the tube is denoted by v and p denotes the fluid pressure in the tube. A finite volume discretization of Eq. (3.44) and Eq. (3.45) yield

$$\frac{\Delta x}{\Delta t}\left(a_i - a_i^n\right) + v_{i+1/2}a_{i+1/2} - v_{i-1/2}a_{i-1/2} - \frac{\alpha}{\rho_f}\left(p_{i+1} - 2p_i + p_{i-1}\right) = 0, \tag{3.46}$$

$$\frac{\Delta x}{\Delta t}\left(v_i a_i - v_i^n a_i^n\right) + v_i v_{i+1/2}a_{i+1/2} - v_{i-1}v_{i-1/2}a_{i-1/2} \frac{1}{2\rho_f}\left(a_{i+1/2}(p_{i+1} - p_i) + a_{i-1/2}(p_i - p_{i-1})\right) = 0, \tag{3.47}$$

with $\alpha = a_0/(v_0 + \Delta x/\Delta t)$ a pressure stabilization term. The inlet velocity

$$v_{in} = v_0 + \frac{v_0}{10}\sin^2(\pi n\tau) \quad \text{with} \quad \tau = \frac{v_0\Delta t}{L}, \tag{3.48}$$

where n is the current time level.

3.4.3. Space mapping function

The low fidelity model can be constructed using a coarser space discretization than the high fidelity model. For the low fidelity model we use $N_v = N_H$ with $\Delta x = H$ while for the high fidelity model we use $N_v = N_h$ with a smaller $\Delta x = h$. We then have $\mathbf{z} \in \mathbb{R}^{N_H}$ and $\mathbf{x} \in \mathbb{R}^{N_h}$ with $N_H < N_h$. The space mapping function can be chosen as

$$\mathcal{P}(\mathbf{x}) = I_H^h \cdot \arg\min_{\mathbf{z}} ||\widetilde{\mathcal{R}}(\mathbf{z}) - I_h^H \cdot \mathcal{R}(\mathbf{x})||, \tag{3.49}$$

where the prolongation operator is denoted by $I_H^h : \mathbb{R}^{N_H} \to \mathbb{R}^{N_h}$ and the restriction operator by $I_h^H : \mathbb{R}^{N_h} \to \mathbb{R}^{N_H}$. Although the space mapping function in Eq. (3.49) is a perfect mapping the function is surjective in the region of interest since high frequency components in the fine model residual $\mathcal{R}$ are filtered out by the restriction operator I_h^H such that it happens that $\mathcal{P}(\hat{\mathbf{x}}) = \mathbf{z}^*$ with $\hat{\mathbf{x}}$ close but not identical

to the fine model solution $\mathbf{x}^*$. The high frequency components of the error $\epsilon = ||\hat{\mathbf{x}} - \mathbf{x}^*||$ cannot be removed by the ASM algorithm as a consequence. It was shown in [13] that standard subiteration provides a good smoother for multi-grid methods. A decision criterium to switch to the smoothing algorithm during the coupling iterations is proposed in [14] where a similar problem was observed with the use of multi-level acceleration and Quasi-Newton acceleration for partitioned strongly coupled fluid-structure interaction. The idea is to split the high fidelity model residual in a part corresponding to the coarse grid and a part corresponding to the fine grid

$$\mathcal{R} = \mathcal{R}_c + \boldsymbol{\delta}_f, \tag{3.50}$$

with $\mathcal{R}_c = I_H^h \cdot I_h^H \cdot \mathcal{R}$ and $\boldsymbol{\delta}_f$ the high frequency components of the residual. When performing space mapping iterations we have $\mathcal{R}_c \to \mathbf{0}$, while $\mathcal{R} \to \boldsymbol{\delta}_f$ and $\mathbf{x} \to \hat{\mathbf{x}}$ with $\hat{\mathbf{x}} \neq \mathbf{x}^*$. When this happens we switch to the smoothing algorithm, e.g. Gauss-Seidel or Quasi-Newton iterations in order to remove high frequency components of the error. The decision to switch to the smoothing algorithm is then based on

$$D = \frac{||\mathcal{R}_c||}{||\boldsymbol{\delta}_f||} = \frac{||\mathcal{R}_c||}{||\mathcal{R} - \mathcal{R}_c||}, \tag{3.51}$$

which measures the ratio between the coarse and high frequency components of the fine model residual. When $D = 1$ the coarse residual and high frequency components are of the same order. When $D \leq 1$ it is not useful anymore to continue the ASM algorithm, making it is necessary to smooth the error with the smoothing algorithm.

3.4.4. Numerical experiments

The physical parameters of this test case are collected in table 3.9. The high fidelity model consists of a finite volume discretization with 250 cells and the low fidelity model discretization uses 80 cells.

Description	symbol	value
Constant fluid density	ρ	1
Youngs modulus	E	1
Tube wall thickness	h	$\sqrt{\frac{4}{\pi}}$
Tube length	L	1

Table 3.9: Physical parameters

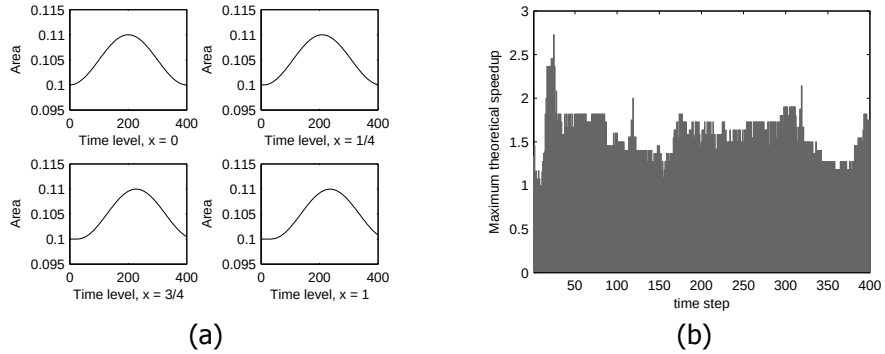
Other numerical parameters are collected in table 3.10.

Description	symbol	value
Initial fluid velocity	v_0	0
Initial fluid pressure	p	0
Switch criterion	D	1
Finite Volumes (course/fine)	N_v	$80 \cup 250$
Number of time steps	N_t	400
Time step	Δt	P/N_t
Outer tolerance	ϵ_I	$1 \cdot 10^{-9}$
Inner tolerance	ϵ_s	$1 \cdot 10^{-12}$

Table 3.10: Initial conditions and numerical parameters

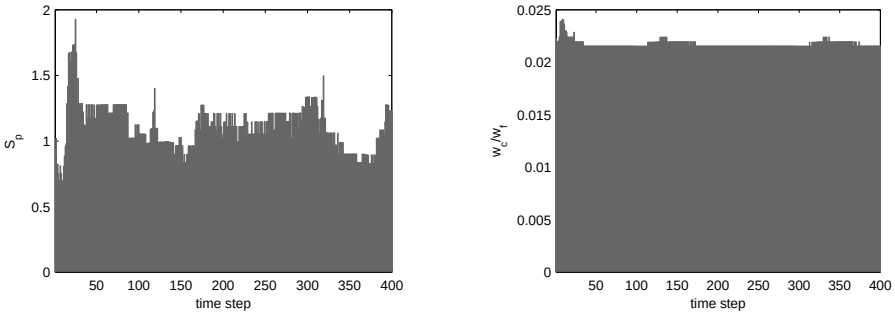
In order to demonstrate the effectiveness of space mapping when small time steps are used we choose 400 time steps per period using a simple backward Euler time-integration scheme. The tube area as a function of time at the locations $x = 0$, $x = 1/4$, $x = 3/4$ and $x = 1$ is given in Figure 3.20a. As can be seen the area of the tube changes gradually with the inlet velocity of the tube. The maximum theoretical speedup of the ASM-ILS algorithm - the speedup that would be achieved if the cost of evaluating the low fidelity model would be zero - is plotted in Figure 3.20b. The maximum theoretical speedup is larger than 1 for all time steps of the simulation which means that solving $\mathcal{P}(\mathbf{x}) - \mathbf{z}^*$ indeed results in a root-finding problem which can be solved faster. The actual coarse model work is however not negligible in this particular test case since there is no exterior domain and the test case is only 1-D. In order to demonstrate the sensitivity of the speedup with respect to the coarse to fine work ratio, Figure 3.21a and Figure 3.21b show the estimated speedup if the coarse to fine work ratio would have been in the order of a few percent.

Figure 3.22a and Figure 3.22b show the decision value D and the residual convergence plot at time step $n = 3$. In this particular time step it can be seen that it is necessary to switch to the QN-ILS algorithm after 7 iterations. A jump of the residual norm is present in the ASM-ILS algorithm directly after switching but the ASM-ILS algorithm still converges faster than the QN-ILS algorithm. Figure 3.23a and Figure 3.23b show the same plots for time step $n = 200$. It can be seen that it is not necessary to switch in this case since the ASM-ILS algorithm converges to the high fidelity solution.



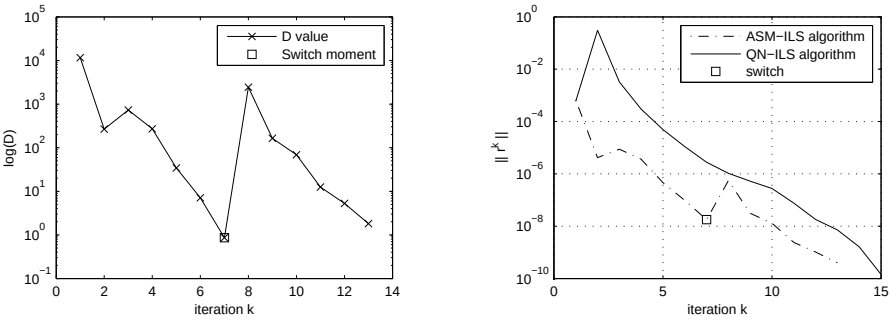
(a) (b)

Figure 3.20: (a) Tube area as a function of time, $\Delta t = P/400$. (b) Maximum theoretical speedup.



(a) (b)

Figure 3.21: (a) Estimated speedup per time step. (b) Average work ratio per time step.



(a) (b)

Figure 3.22: (a) D value of the ASM-ILS algorithm at time n = 3. (b) Interface residual convergence at time n = 3.

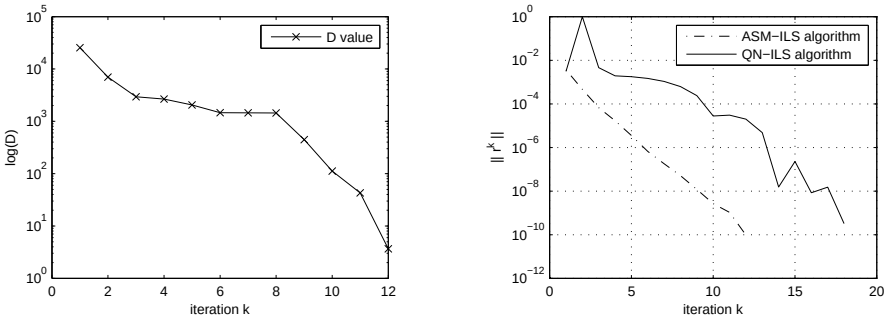


Figure 3.23: (a) D value of the ASM-ILS algorithm at time n = 200. (b) Interface residual convergence at time n = 200.

3.5. Conclusions

Space-mapping is a general multi-fidelity technique that can be applied to a wide variety of engineering problems. The purpose of this contribution is:

1. To provide a framework for the application of the space-mapping technique to partitioned FSI problems in order to speed up high fidelity simulations.

2. To demonstrate the application of the Aggressive Space-Mapping technique to various academic fluid-structure interaction problems.

Aggressive Space-Mapping has been successfully applied to three academic fluid-structure interaction problems of increasing complexity in order to speed up the partitioned algorithm. This is achieved by the definition of a space-mapping function on the FSI interface together with a proper way of constructing the space-mapping Jacobian using input/output information. For the test cases under consideration we found that:

1. The estimated and observed speedup of the Aggressive Space-Mapping algorithm with respect to the Quasi-Newton algorithm is larger than 1 for most test cases considered.

2. The influence of the time step on the achieved speedup is large compared to other parameters. When compressible flows are studied the speedup is close to 1 for small time steps while for larger time steps the speedup can rise to 1.5. For incompressible flows the fluid-structure interaction is much stronger and for this reason higher speedups can be obtained.

The space-mapping framework allows for the design of new coupling algorithms that – using the information of simplified fluid models – can significantly speed up

partitioned strongly coupled FSI simulations. Off the shelf solvers can be reused since the space-mapping technique is solely based on the use of input/output information. This makes the framework especially useful in an industrial environment.

References

[1] S. Piperno, C. Farhat, and B. Larrouturou, *Partitioned procedures for the transient solution of coupled aeroelastic problems part i: Model problem, theory and two-dimensional application,* Computer Methods in Applied Mechanics and Engineering **124**, 79 (1995).

[2] E. Lefrançois and J. P. Boufflet, *An introduction to Fluid-Structure interaction: Application to the piston problem,* SIAM Review **52**, 747 (2010).

[3] E. H. van Brummelen and R. d. Borst, *On the nonnormality of subiteration for a fluid-structure-interaction problem,* SIAM J. Sci. Comput. **27**, 599 (2005).

[4] A. H. Zuijlen, S. Bosscher, and H. Bijl, *Two level algorithms for partitioned fluid-structure interaction computations,* Computer Methods in Applied Mechanics and Engineering **196**, 1458 (2007).

[5] A. H. Zuijlen, *Fluid-structure interaction simulations*, Ph.D. thesis, Delft University of Technology (2006).

[6] S. Piperno and C. Farhat, *Partitioned procedures for the transient solution of coupled aeroelastic problems - part ii: energy transfer analysis and three-dimensional applications,* Computer Methods in Applied Mechanics and Engineering **190**, 3147 (2001).

[7] T. J. R. Hughes, *The Finite Element Method* (Prentice-Hall, 1987).

[8] C. V. Verhoosel, T. P. Scholcz, S. J. Hulshoff, and M. A. Gutiérrez, *Transient stability analysis of an aeroelastic problem with random fluid and structure properties,* in *Proceedings of the 10th International Conference on Structural Safety and Reliability (ICOSSAR10)* (2009) pp. 377–384, 13-17 September 2009, Osaka, Japan.

[9] E. H. van Brummelen, *Added mass effects of compressible and incompressible flows in fluid-structure interaction,* Journal of Applied Mechanics **76**, 021206 (2009).

[10] P. Causin, J. F. Gerbeau, and F. Nobile, *Added-mass effect in the design of partitioned algorithms for fluid–structure problems,* Computer Methods in Applied Mechanics and Engineering **194**, 4506 (2005).

[11] J. Degroote and J. Vierendeels, *Multi-solver algorithms for the partitioned simulation of fluid-structure interaction,* Computer Methods in Applied Mechanics and Engineering **200**, 2195 (2011).

[12] J. Degroote, P. Bruggeman, R. Haelterman, and J. Vierendeels, *Stability of a coupling technique for partitioned solvers in fsi applications,* Computers & Structures **86**, 2224 (2008).

[13] E. H. van Brummelen, *Partitioned iterative solution methods for fluid-structure interaction,* International Journal for Numerical Methods in Fluids **65**, 3 (2011).

[14] J. J. Kreeft, M. Weghs, A. H. Zuijlen, and H. Bijl, *Multi-level and quasi-newton acceleration for strongly coupled partitioned fluid-structure interaction,* in *Proceedings of the 4th International Conference on Computational Methods for Coupled Problems in Science and Engineering (ECCOMAS 2011)* (2011) 20-22 June 2011, Kos Island, Greece.

II

Multi-fidelity analysis for uncertainty quantification

4

High-dimensional meta-models for uncertainty quantification

This chapter contains a short note on high-dimensional meta-models and the problems that occur when a large number of dimensions are considered. A theoretical speedup can be obtained when gradients are included in the construction of a meta-model. This is the subject of section 4.1. In Section 4.2 we define a meta-modeling technique and in Section 4.3 it is shown how gradient information can be used to improve the meta-modeling performance. A performance metric is then developed in Section 4.4 including a theoretical result of the improved performance. This metric is then used in Chapter 5 and Chapter 6 to assess whether the theoretical result is found in practice.

4.1. Theoretical speedup

Building high-dimensional metamodels of expensive computational codes is necessary for efficient optimization and uncertainty quantification [1]. However the computational cost of sampling parameter spaces scales exponentially with the number of parameters d—the well-known *curse of dimensionality*. One possibility to mitigate this effect in the case of computational codes is to exploit the availability of adjoint derivatives [2]. A (complete) adjoint of a code can provide derivatives of a scalar code output (e.g. aerodynamic drag, C_{D}) with respect to an arbitrary number of input parameters at a fixed cost; usually approximately the same cost as a single run of the original code. Assuming each derivative is as informative to the surrogate as the code output itself, we obtain $d+1$ scalar data at a cost of ≈ 2 code runs, and should expect a speed-up compared to sampling without derivatives by

a factor of [3]:

$$S \approx \frac{1}{2}(1+d). \tag{4.1}$$

That is, for a given level of metamodel error, the number of required runs would be reduced with a factor S by using adjoint derivatives. As a result the use of gradient information has been indicated as a promising approach in a recent metamodelling review for optimization [1], and uncertainty quantification [4, 5].

However the theoretical performance of Eq. (4.1) is seldom achieved in practice, and instead it is observed that adding derivative information often *degrades* metamodel accuracy [2, 6]. In chapter 5 and 6 it is demonstrated on two simple numerical cases that the expected performance can be achieved, but only if we properly account for the inherent noise (or error) in the derivative information.

Numerical noise is an issue in any surrogate modeling approach [7, 8], but adjoint derivatives in CFD are particularly susceptible to errors [2, 9]. In the case of continuous adjoints, gradients will only approximate the gradient of the discrete code. In most discrete RANS adjoints, the turbulence is frozen [2], resulting in unpredictable errors. Even in the rare and ideal case of a *fully exact* discrete adjoint, Giles et al. show [9] that the adjoint gradient may be corrupted by very fine-scale numerical artifacts in the code output, and thereby not represent the overall trend. Error estimates do not currently exist for any of these errors.

In this work we model the gradient errors as Gaussian noise, that is: unbiased, uncorrelated, and of constant standard deviation (in the parameter space). This standard deviation we obtain from a maximum-likelihood estimation (MLE). Our noise-model for the error is certainly imperfect. Errors resulting from approximate adjoints will be biased and correlated. Errors due to the effects in [9] although uncorrelated, are likely to be biased. We consider our error model an acceptable compromise between a more complete error-model with many hyperparameters, and not modelling the error at all.

4.2. Kriging

A Kriging metamodel [10–12] is used in order to investigate these issues. The statistical foundation of Kriging provides a natural treatment of noise in observations. In a Bayesian framework, the Kriging predictor is given by [13, 14]:

$$E(\mathbf{x}|\mathbf{y}) = \boldsymbol{\mu} + PH'(R + HPH')^{-1}(\mathbf{y} - H\boldsymbol{\mu}), \tag{4.2}$$

with quantity of interest $\mathbf{x}$, observations $\mathbf{y}$, drift $\boldsymbol{\mu}$, covariance matrix P, observation error covariance matrix R, and observation matrix H.

4.3. Gradient Enhanced Kriging

The extension to the use of derivatives, known as Gradient-Enhanced Kriging (GEK), is straightforward: The gradient information can be introduced as covariables through $\mathbf{y}_c$ and H_c, while the gradient noise is expressed in the error covariance matrix [14, Eq. 12]:

$$R_c = \begin{pmatrix} \epsilon^2 \mathbf{I} & 0 \\ 0 & \epsilon_\nabla^2 \mathbf{I} \end{pmatrix} \tag{4.3}$$

with ϵ the standard deviation of the noise in the objective function, and ϵ_∇ the standard deviation of the noise in the gradient of the objective function. The GEK predictor is given by a straightforward extension of (4.2):

$$E(\mathbf{x}|\mathbf{y}_c) = \boldsymbol{\mu}_c + P_c H_c' (R_c + H_c P_c H_c')^{-1} (\mathbf{y}_c - H_c \boldsymbol{\mu}_c), \tag{4.4}$$

which requires a special form of the matrices R_c and P_c, as explained in [14].

GEK has been applied to a range of problems, for example: Rumpfkeil et al. use [15] a Kriging metamodel with gradient and Hessian information, without accounting for noise; Lockwood and Anitescu use [16] Universal Kriging with gradient information, without accounting for noise; and Dwight and Han use [6] indirect GEK, also without accounting for noise.

4.4. Observed speedup

The observed speedup is the speedup S_p obtained by using GEK instead of Kriging. In order to reach a certain accuracy of the Kriging response surface, we need N_K solves. To estimate the number of required GEK solves N_GEK, one could assume that the total amount of required data $(1+d)N_\mathrm{GEK} = N_\mathrm{K}$ depends on the number of random variables d (i.e. that the gradients are as informative as the values), such that the speedup becomes:

$$\begin{aligned} S_p &= \frac{\mathrm{cost}_\mathrm{K}}{\mathrm{cost}_\mathrm{GEK}} \\ &\approx \left(\frac{t_\mathrm{val}}{t_\mathrm{val} + t_\mathrm{grad}} \right) (1+d)\,, \end{aligned} \tag{4.5}$$

In practice it holds that $t_\mathrm{val} \neq t_\mathrm{grad}$ such that the observed speedup based on CPU time differs from the theoretical speedup in Eq. (4.1). Calculating the gradients with the adjoint method is usually slightly more expensive than calculation of the objective function, hence $t_\mathrm{grad}/t_\mathrm{val} > 1$, depending on the problem and the platform used for the calculations. In order to obtain results that are platform independent define

$$S_p = \frac{N_\mathrm{K}}{N_\mathrm{GEK}} \quad \text{and} \quad \eta_{CPU} = \frac{t_\mathrm{val} + t_\mathrm{grad}}{t_\mathrm{val}}. \tag{4.6}$$

Testing whether the theoretical speedup in Eq. (4.1) holds now breaks down into two hypothesis tests:

1. Are gradients as informative as values for the metamodel?

$$\frac{N_{\mathrm{K}}}{N_{\mathrm{GEK}}} \overset{\mathrm{hyp}}{=} (1+d) \tag{4.7}$$

2. Is the cost of an adjoint solve equal to the cost of a primary solve?

$$\frac{t_{\mathrm{val}} + t_{\mathrm{grad}}}{t_{\mathrm{val}}} \overset{\mathrm{hyp}}{=} 2 \tag{4.8}$$

The theoretical speedup is attained if and only if both hypotheses are true. In practice we test whether they are sufficiently satisfied.

References

[1] F. A. C. Viana, T. W. Simpson, V. Balabanov, and V. Toropov, *Metamodeling in multidisciplinary design optimization: How far have we really come?* AIAA Journal **52**, 670 (2014).

[2] R. Dwight and J. Brezillon, *Effect of approximations of the discrete adjoint on gradient-based optimization,* AIAA Journal **44**, 3022 (2006).

[3] J. H. S. de Baar, T. P. Scholcz, C. V. Verhoosel, R. P. Dwight, A. H. van Zuijlen, and H. Bijl, *Efficient uncertainty quantification with gradient-enhanced kriging: Applications in FSI,* Eccomas Vienna (2012).

[4] T. Oden, R. Moser, and O. Ghattas, *Computer predictions with quantified uncertainty (Part 1),* SIAM News **43** (2010).

[5] T. Oden, R. Moser, and O. Ghattas, *Computer predictions with quantified uncertainty (Part 2),* SIAM News **43** (2010).

[6] R. P. Dwight and H. Zhong-Hua, *Efficient uncertainty quantification using gradient-enhanced kriging,* 50th Structures, Structural Dynamics, and Materials Conference, 4-7 May 2009, Palm Springs, California .

[7] A. I. J. Forrester, A. J. Keane, and N. W. Bressloff, *Design and analysis of "noisy" computer experiments,* AIAA Journal **44**, 2331 (2006).

[8] A. I. J. Forrester, A. Sóbester, and A. J. Keane, *Engineering Design via Surrogate Modelling, A Practical Guide* (Wiley, 2008).

[9] M. B. Giles, M. C. Duta, J. D. Muller, and N. A. Pierce, *Algorithm developments for discrete adjoint methods,* AIAA Journal **41**, 198 (2003).

[10] G. Matheron, *Principles of Geostatistics,* Economic Geology **58**, 1246 (1963).

[11] L. S. Gandin, *Objective analysis of meteorological fields: Gidrometeorologicheskoe Izdatel'stvo (GIMIZ), Leningrad* (Translated by Israel Program for Scientific Translations, Jerusalem, 1965).

[12] J. Sacks, W. J. Welch, T. Mitchell, and H. P. Wynn, *Design and analysis of computer experiments,* Statistical Science **4**, 409 (1989).

[13] C. K. Wikle and L. M. Berliner, *A Bayesian tutorial for data assimilation,* Physica D: Nonlinear Phenomena **230**, 1 (2007).

[14] J. H. S. de Baar, R. P. Dwight, and H. Bijl, *Improvements to Gradient-Enhanced Kriging using a Bayesian Interpretation,* International Journal for Uncertainty Quantification **4**, 205 (2014).

[15] M. P. Rumpfkeil, W. Yamazaki, and D. J. Mavriplis, *Uncertainty Analysis Utilizing Gradient and Hessian Information,* in *Computational Fluid Dynamics 2010*, edited by A. Kuzmin (Springer Berlin Heidelberg, 2011) pp. 261–268.

[16] B. A. Lockwood and M. Anitescu, *Gradient-enhanced universal kriging for uncertainty propagation,* (2010).

5

Panel divergence problem

In this chapter, Uncertainty Quantification (UQ) is used to propagate structural uncertainties – which might arise during e.g. manufacturing or operation – through a fluid-structure interaction (FSI) problem [1]. This requires an UQ method, which generally suffers from the 'curse of dimensionality': the uncertainty propagation becomes increasingly computationally expensive when the number of random variables increases [1, 2].

The objective of this research is to mitigate the curse of dimensionality by including not only values but also adjoint-based gradients in the UQ process. This is possible when we apply Gradient-Enhanced Kriging (GEK) [3–7], which we will present as an extension of the perturbation method (also known as 1^{st} order moment method). We compare the results of Kriging and GEK, and consider both the estimated and observed speedup.

Our scope of coverage is limited to a number of random variables ranging from 1 to 16. As a second limitation, the current FSI problem exhibits a highly linear response, such that the effect of adding gradient information (apart from the gradient information in the *prior*, i.e. the gradient information obtained at the first evaluation) is only seen when we are interested in very high accuracies.

5.1. Primary problem

Following [8] we consider the interaction of a fluid and a flexible panel of length h suspended between two rigid walls. See Figure 5.1 for a schematic representation of the problem setup. The deflection $w(x)$ of the panel is governed by the Euler-

[1] This chapter is based on the conference paper: **J. H. S. de Baar, T. P. Scholcz, C. V. Verhoosel, R. P. Dwight, A. H. van Zuijlen and H. Bijl**, *Efficient uncertainty quantification with gradient-enhanced kriging: applications in FSI*, 6th European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS 2012, Vienna (2012).

Bernoulli beam equation and boundary conditions:

$$\begin{cases} \frac{\partial^2}{\partial x^2}\left[\frac{\tilde{E}(x)I}{1-\nu^2}\frac{\partial^2 w}{\partial x^2}\right] = -\Delta p(x) \quad \text{on } \Gamma_p, \\ w(\pm\frac{h}{2}) = \frac{\partial w}{\partial x}(\pm\frac{h}{2}) = 0, \end{cases} \tag{5.1}$$

where I and ν are the moment of intertia and Poisson's ratio, respectively. The modulus of elasticity $\tilde{E}(x)$ is a stationary lognormal random field, parametrized by the mean μ_E, standard deviation σ_E, and autocorrelation function $\rho(x_1, x_2)$. Note that the tilde indicates a random variable. The log normal random field $\tilde{E}(x)$ is discretized into standard Gaussian random variables, $\tilde{\mathbf{z}} = [\tilde{z}_1, \ldots, \tilde{z}_m]$, by the Karhunen-Loeve expansion [9]:

$$\tilde{E}(x) = E(x, \tilde{\mathbf{z}}) = \frac{\mu_E}{\sqrt{1+V_E^2}} \prod_{i=1}^{m} \exp(\sqrt{\zeta_i} r_i(x) \tilde{z}_i), \tag{5.2}$$

where $V_E = \frac{\sigma_E}{\mu_E}$. The Karhunen-Loeve eigenvalues ζ_i and eigenfunctions r_i are determined numerically, see [9] for details. Interaction with the fluid occurs through the pressure difference $\Delta p(x)$ between the top and the bottom of the plate. The boundary value problem (5.1) is discretized using the finite element method with Hermitian shape functions, such that the nodal displacements and rotations are used as coefficients for the approximate deformation of the panel.

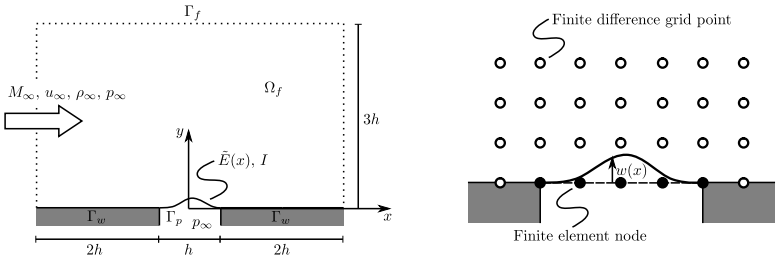


Figure 5.1: Schematic representation of the panel divergence problem

The fluid motion is modeled by the steady linearized potential equation, for which the horizontal and vertical velocity fields are related to the disturbance potential $\phi(x, y)$ through:

$$u = u_\infty + \frac{\partial \phi}{\partial x}, \qquad v = \frac{\partial \phi}{\partial y}, \tag{5.3}$$

with u_∞ the free stream horizontal velocity. In the subsonic case, i.e. when the free stream Mach number is smaller than $M_\infty = 1$, the disturbance potential field is

governed by the (primary) elliptic boundary value problem:

$$\begin{cases} \left(1 - M_\infty^2\right) \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 & \text{in } \Omega_f, \\ \phi(x,y) = 0 & \text{on } \Gamma_f, \\ v(x,y) = 0 & \text{on } \Gamma_w, \\ v(x,y) = u_\infty \frac{\partial w}{\partial x} & \text{on } \Gamma_p, \end{cases} \tag{5.4}$$

where coupling with the panel is established through the kinematic boundary condition at the panel surface, Γ_p. The pressure difference between the top and bottom of the panel, which serves as the right-hand-side in the boundary value problem for the panel (5.1), is expressed in terms of the disturbance potential field as:

$$\Delta p = -\rho_\infty u_\infty \frac{\partial \phi}{\partial x}, \tag{5.5}$$

where ρ_∞ is the free stream fluid density. The fluid boundary value problem (5.4) is discretized using the finite difference method. The finite difference grid is chosen such that it conforms with the finite element mesh of the panel.

In this contribution we consider the monolithically coupled aeroelasticity problem, which – using the discretizations mentioned above – results in the system:

$$\mathbf{K}(M_\infty, \tilde{\mathbf{z}})\mathbf{a} = \begin{bmatrix} \mathbf{K}^{pp}(\tilde{\mathbf{z}}) & \mathbf{K}^{pf}(M_\infty) \\ \mathbf{K}^{fp}(M_\infty) & \mathbf{K}^{ff}(M_\infty) \end{bmatrix} \begin{bmatrix} \mathbf{a}^p \\ \mathbf{a}^f \end{bmatrix} = \mathbf{0}, \tag{5.6}$$

where $\mathbf{a}$ is the composite vector of fluid and structure degrees of freedom. The boundary for occurence of aeroelastic divergence, a fluid-structure instability which can occur at subsonic speeds [8], is related to the presence of nontrivial solutions for $\mathbf{a}$, which only exist when the stiffness matrix $\mathbf{K}$ in (5.6) is singular. With the smallest in magnitude eigenvalue of the matrix $\mathbf{K}$ denoted as:

$$|\lambda_1| = |\lambda_1|(M_\infty, \tilde{\mathbf{z}}), \tag{5.7}$$

the stability boundary M_{div}, i.e. the Mach number which separates the stable from the unstable regime, is governed by:

$$|\lambda_1|(M_{\text{div}}, \tilde{\mathbf{z}}) = 0. \tag{5.8}$$

The divergence Mach number is computed using a Newton-Rhapson procedure after the computation of an initial approximation from a plot of the real and imaginary parts of λ_1, see [8] for details.

5.2. Adjoint problem

The objective of the current work is to investigate the increased efficiency of response surface based uncertainty quantification, as a result of including gradient information. Therefore, we would like to compute the gradients of M_{div} with respect to the random variables $\tilde{z}_i$, in such a way that the relative computational cost

of obtaining these gradients is independent of the number of random variables. Presently, we compute the gradients with an adjoint approach.

Differentiating equation (5.8) with respect to the random variable $\widetilde{z}_i$ yields [**?**]:

$$\frac{d|\lambda_1|}{d\widetilde{z}_i} = \frac{\partial|\lambda_1|}{\partial M_{\text{div}}}\frac{\partial M_{\text{div}}}{\partial \widetilde{z}_i} + \frac{\partial|\lambda_1|}{\partial \widetilde{z}_i} = 0 \quad \text{for} \quad i = 1, ..., m, \tag{5.9}$$

such that the gradient of the divergence Mach number is given by:

$$\frac{\partial M_{\text{div}}}{\partial \widetilde{z}_i} = -\left(\frac{\partial|\lambda_1|}{\partial M_{\text{div}}}\right)^{-1}\frac{\partial|\lambda_1|}{\partial \widetilde{z}_i} \quad \text{for} \quad i = 1, ..., m, \tag{5.10}$$

where the smallest in magnitude eigenvalue λ_1 is found by solving:

$$\det|\mathbf{K} - \lambda\mathbf{I}| = 0. \tag{5.11}$$

5

The right and left eigenvectors $\mathbf{u}_1$ and $\mathbf{v}_1$ corresponding to λ_1 satisfy the system:

$$(\mathbf{K} - \lambda_1\mathbf{I})\mathbf{u}_1 = \mathbf{0}, \tag{5.12}$$

and the *adjoint* system:

$$(\mathbf{K}^T - \lambda_1\mathbf{I})\mathbf{v}_1 = \mathbf{0}. \tag{5.13}$$

Taking the derivative of (5.12) with respect to a parameter θ we find:

$$\frac{\partial\lambda_1}{\partial\theta}\mathbf{u}_1 = \frac{\partial\mathbf{K}}{\partial\theta}\mathbf{u}_1 + (\mathbf{K} - \lambda_1\mathbf{I})\frac{\partial\mathbf{u}_1}{\partial\theta}, \tag{5.14}$$

which we multiply with $\mathbf{v}_1^T$, such that the second term in (5.14) drops. The eigenvalue gradient with respect to a parameter θ is now given by:

$$\frac{\partial\lambda_1}{\partial\theta} = \frac{\mathbf{v}_1^T\frac{\partial\mathbf{K}}{\partial\theta}\mathbf{u}_1}{\mathbf{v}_1^T\mathbf{u}_1}, \tag{5.15}$$

where θ is either $\widetilde{z}_i$ or M_{div}. In (5.15) the matrix gradients $\frac{\partial\mathbf{K}}{\partial\widetilde{z}_i}$ are given by:

$$\frac{\partial\widetilde{\mathbf{K}}}{\partial\widetilde{z}_i} = \frac{\partial\mathbf{K}}{\partial\widetilde{E}}\frac{\partial\widetilde{E}}{\partial\widetilde{z}_i} \quad \text{and} \quad \frac{\partial\widetilde{E}}{\partial\widetilde{z}_i} = \sqrt{\zeta_i}r_i(x)E(x,\widetilde{\mathbf{z}}) \quad \text{for} \quad i = 1, ..., m. \tag{5.16}$$

Since we have to solve the system (5.12) repeatedly to obtain (5.8), whereas we have to solve the adjoint system (5.13) only once, we expect that for this particular problem the computational cost of obtaining the gradients is significantly lower than that of obtaining the value M_{div}.

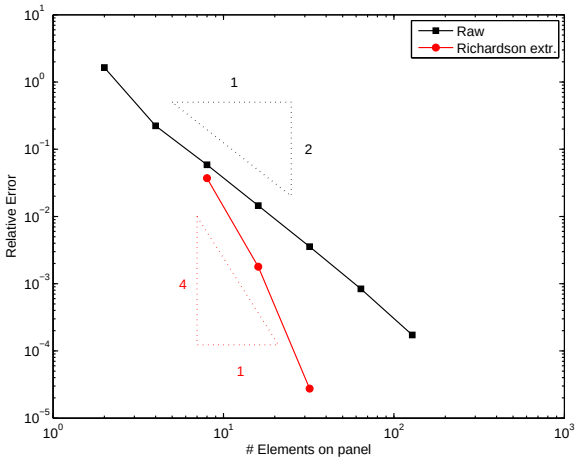


Figure 5.2: Grid convergence of M_{div} at a random location in a 16-random variable space

5.3. Verification and error estimation

Figure 5.2 illustrates the second order grid convergence for the raw (i.e. single grid, unextrapolated) results. However, since the results are in the exponential region for grids with 4 or more elements on the panel, we use a Richardson extrapolation based on three grids. The Richardson extrapolation is given by [10]:

$$M_{\text{extrap}} = M_{\text{fine}} + \frac{M_{\text{fine}} - M_{\text{medium}}}{r^p - 1}, \tag{5.17}$$

with $r = 2$ the growth factor of the consecutive grids and [10]:

$$p = \log\left(\frac{M_{\text{coarse}} - M_{\text{medium}}}{M_{\text{medium}} - M_{\text{fine}}}\right) / \log(r), \tag{5.18}$$

the observed rate of convergence. This leads to a fourth order grid convergence, as illustrated in Figure 5.2. The coarse, medium, and fine grid will have 5, 10, and 20 elements on the panel in the following sections. These grids correspond to 8, 18, and 38 structural degrees of freedom and 416, 1581, and 6161 fluid degrees of freedom.

5.4. Response surfaces

In this section we will discuss several response surface techniques. Since the response of the present problem is fairly linear, the 1st order perturbation method is a very good starting point for approximating the response. We will use this linear response as a *prior*, which we can update progressively with the results of additional computations using Kriging or Gradient-Enhanced Kriging (GEK), until we reach a certain desired accuracy of the response surface.

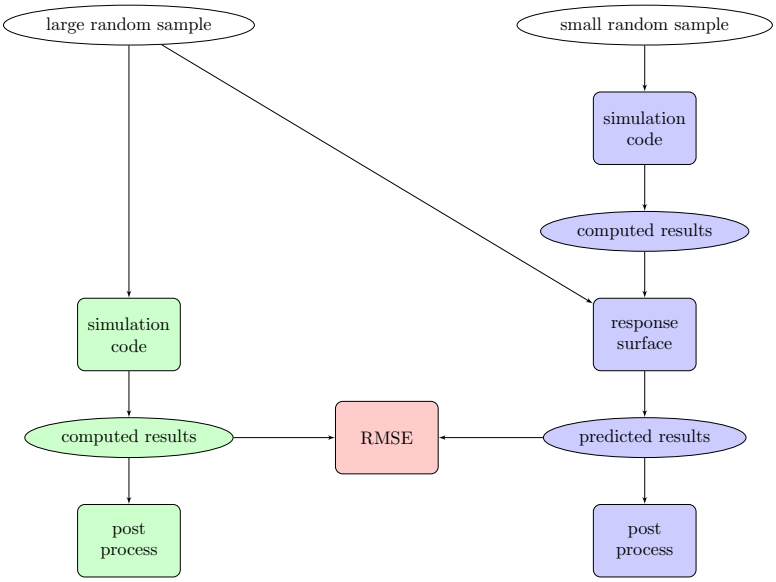


Figure 5.3: Flow chart of a Monte Carlo simulation (left-hand side, in green) and the present response surface technique (right-hand side, in blue). The response surface accuracy is given by the Root Mean Squared Error (RMSE) (middle, in red)

5.4.1. Response Surface Techniques

In order to introduce the response surface approach, first we would like to briefly discuss Monte Carlo sampling [11], a widely applied method for Uncertainty Quantification (UQ). The Monte Carlo approach is illustrated in Figure 5.3 on the left-hand side (green), and consists of the following steps: (1) from the random input variable space we draw a *large* number of samples (i.e. realizations, in the present case $10,000$), which are distributed according to the input variable probability density function (pdf); (2) we run the code for each of these samples; and (3) we post process the *computed* results to obtain the desired statistics of our system. In general, this approach is considered to be very robust (in the sense that 1. it is applicable to arbitrary pdf's and 2. the sample size for a given confidence level is independent of the dimension of the random variable space) as well as rather costly.

The objective of a response surface technique is to obtain the same statistics at a lower cost, by reducing the number of code runs. The present approach is illustrated in Figure 5.3 on the right-hand side (blue), and consists of the following steps: (1) from the random input variable space we draw a *small* number of samples, which are distributed according to the input variable pdf; (2) we run the code for each of these samples; (3) we develop a response surface which acts as a surrogate of the code response; (4) from the random input parameter space we now draw a *large* number of samples, which are distributed according to the input parameter pdf; (5) we use the response surface to *predict* the code response at these inputs; and (6) we post process the *predicted* results to obtain the desired statistics of our system.

The challenge is to develop an efficient response surface technique, which gives accurate predictions of the code output at a relatively low cost. In this section, we will introduce several response surface techniques. In Section 5.5 will then consider the accuracy of the response surfaces by evaluating the Root Mean Squared Error (RMSE) of the predictions, as indicated in the middle (in red) in Figure 5.3.

5.4.2. Perturbation Method

The 1^{st} order perturbation method approximates the response of the code with a first order polynomial. To arrive at this polynomial, one computes M_{div} and it's gradients in the center of the random variable space. Because the present response is fairly linear the perturbation method is quite accurate, in the results section we will see that the RMSE of the response is in the order of 10^{-4}. However, the question remains how to improve on these results by running the code for a larger number of samples.

5.4.3. Perturbation-based prior (ppb)

For Kriging, we use a very simple prior $\boldsymbol{\mu}$ by setting it equal to the mean of the computed results, an approach commonly known as *simple Kriging*.

Since the problem is fairly linear, our next step is to use the 1^{st} order perturbation method as a prior. In this approach, we compute prior values $\boldsymbol{\mu}$ from the perturbation method, after which we update our predictions progressively with newly computed results through the Kriging predictor (4.2). Note that we do not obtain the prior from linear regression, but from the perturbation method.

For Gradient Enhanced Kriging, we use a perturbation-based prior $\boldsymbol{\mu}_c$, followed by progressive updating based on values as well as gradients, this approach can be considered as a full extension of the perturbation method.

5.5. Numerical experiments

In the previous section we have discussed four different response surface techniques: the Perturbation Method, Kriging (no pbp), Kriging (pbp), and Gradient-Enhanced Kriging (pbp). In the present section we will compare the RMSE accuracy of these methods for an increasing number of solves. From (4.5) we expect that when we increase the number of random variables, GEK (pbp) gives an increasing speedup when compared to Kriging (pbp). We will compare the observed speedup with the estimated speedup.

The estimated speedup (4.5) contains the CPU ratio, which we will now determine empirically. We determine the CPU times requires for a solve with gradients and a solve without gradients on a 3.4 GHz Intel Pentium 4 processor. This ratio increases only slightly when we increase the number of random variables, as can be seen in Figure 5.4. This is an important first result, since now we can compute the estimated speedup (4.5).

For each number of random variables, we increase the number of solves while we compute the RMSE accuracy of the response surfaces. This is illustrated for the case of 8 random variables in Figure 5.5. Note that the perturbation method comes

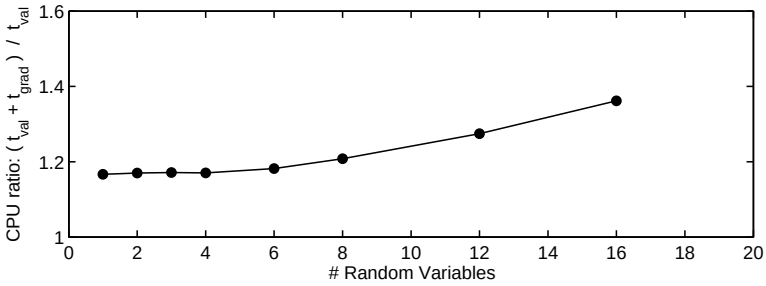


Figure 5.4: The relative cost of computing values and gradients, compared to the cost of computing only values

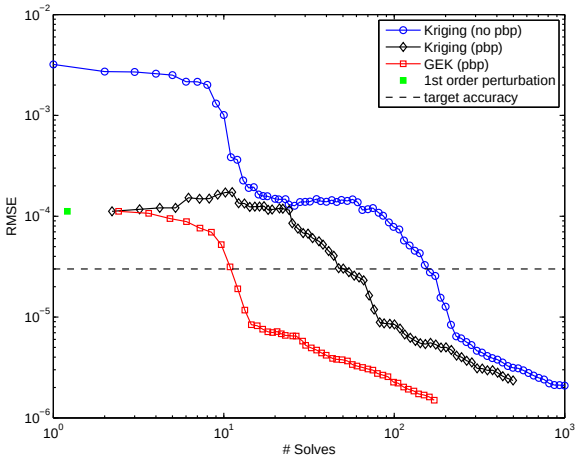


Figure 5.5: When we increase the number of solves progressively, the accuracy of the response surfaces increases

at the cost of roughly 1.2 solves (value and gradients), and that it is not possible to improve it's accuracy by adding more solves. However, Kriging (pbp) and GEK (pbp) emerge as extensions of the Perturbation Method, both improve the accuracy when we add more solves. As expected, even when we consider the increased cost of computing the gradients, GEK (pbp) shows to be more efficient. Kriging (pbp) and GEK (pbp) show typical Kriging convergence: an initial bell shape, followed by a Monte Carlo like tail, where the cut off accuracy of RMSE $\approx 1 \times 10^{-5}$ is probably due to one of the tolerances in the code (which we have not been able to identify yet). The dotted horizontal line in Figure 5.5 corresponds to a target accuracy of RMSE $= 3 \times 10^{-5}$. Figure 5.6 shows the computational cost of reaching this target accuracy for the different response surface methods. This clearly illustrates how the curse of dimensionality is mitigated.

Finally, we consider the speedup we achieve by using GEK (pbp) instead of Kriging (pbp). The estimated and observed speedup are shown in Figure 5.7. Although

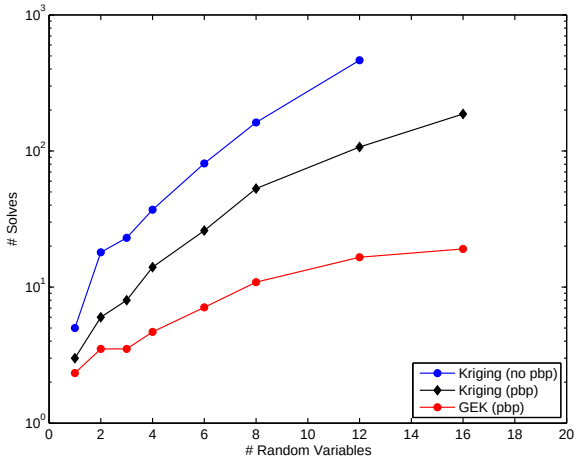


Figure 5.6: Curse of dimensionality: The computational cost of obtaining a target accuracy of RMSE = 3×10^{-5}

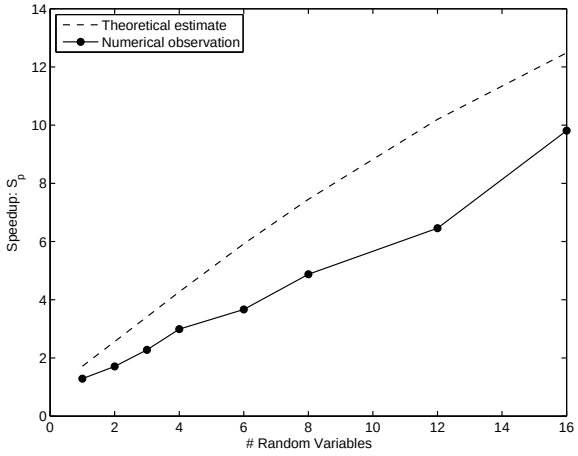


Figure 5.7: The speedup as a result of using gradient information. Estimate from (4.5)

the observed speedup does increase when we increase the number of random variables, it is slightly lower than the estimated speedup.

5.6. Conclusions

We apply Kriging and Gradient-Enhanced Kriging (GEK) as response surface techniques for Uncertainty Quantification (UQ) for the fluid structure interaction panel divergence problem, with up to 16 random variables. The quantity of interest is the critical Mach number, for which divergence occurs. The random variables originate from a random field discretization of the Young's modulus of the panel.

When we consider a certain target accuracy of the response surface, the speedup of GEK as compared to Kriging increases when we increase the number of random variables. This is the case for the estimated as well as for the observed speedup. However, the observed speedup is slightly lower than estimated.

From these results, we are motivated to continue the development of GEK as an efficient tool for UQ of systems with a large number of random variables, for which gradient information is available at reasonable cost.

References

[1] T. Oden, R. Moser, and O. Ghattas, *Computer predictions with quantified uncertainty (Part 1),* SIAM News **43** (2010).

[2] T. Oden, R. Moser, and O. Ghattas, *Computer predictions with quantified uncertainty (Part 2),* SIAM News **43** (2010).

[3] M. D. Morris and D. Y. T. J. Mitchell, *Bayesian design and analysis of computer experiments: Use of derivatives in surface prediction,* Technometrics **35-3**, 243 (1993).

[4] H.-S. Chung and J. J. Alonso, *Using gradients to construct cokriging approximation models for high-dimensional design optimization problems,* AIAA 40th Aerospace Sciences Meeting and Exhibit (2002).

[5] J. Laurenceau and P. Sagaut, *Building efficient response surfaces of aerodynamic functions with kriging and cokriging,* AIAA Journal **46**, 498 (2008).

[6] R. P. Dwight and H. Zhong-Hua, *Efficient uncertainty quantification using gradient-enhanced kriging,* 50th Structures, Structural Dynamics, and Materials Conference, 4-7 May 2009, Palm Springs, California .

[7] J. H. S. de Baar, R. P. Dwight, and H. Bijl, *Improvements to Gradient-Enhanced Kriging using a Bayesian Interpretation,* International Journal for Uncertainty Quantification **4**, 205 (2014).

[8] C. V. Verhoosel, T. P. Scholcz, and S. J. Hulshoff, *Uncertainty and reliability analysis of fluid–structure stability boundaries,* AIAA Journal **47**, 91 (2009).

[9] S. P. Huang, S. T. Quek, and K. K. Phoon, *Convergence study of the truncated karhunen–loeve expansion for simulation of stochastic processes,* International Journal for Numerical Methods in Engineering **52**, 1029 (2001).

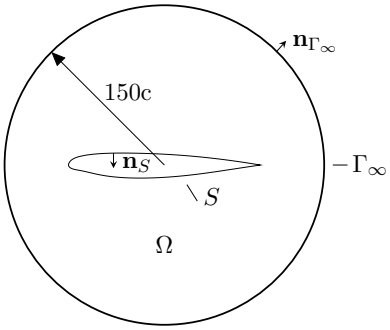
[10] P. J. Roache, *Verification and Validation in Computational Science and Engineering* (Hermosa Publishers, 1998).

[11] N. Metropolis and S. Ulam, *The Monte Carlo Method,* Journal of the American Statistical Association **44**, pp. 335 (1949).

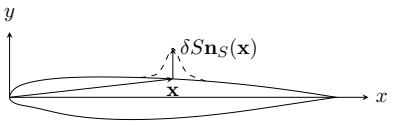
6

FFAST airfoil problem

In this chapter, steady transonic flow over the FFAST (Future Fast Aeroelastic Simulation Technologies) airfoil [1, p. 110] subject to stochastic shape deformations is studied [1]. The onflow conditions are Mach number $M_\infty = 0.8$ and angle of attack $\alpha = 1.25$ degrees. The flow domain is circular with a radius of 150 chord-lengths, see Figure 6.1a.



(a) Schematic of the flow domain.



(b) Airfoil and shape deformation.

Figure 6.1: Schematic of the physical problem

The computational flow domain is denoted by Ω with the outward normals $\mathbf{n}_S$ and $\mathbf{n}_{\Gamma_\infty}$ on the airfoil surface S and far-field boundary Γ_∞ respectively. The airfoil domain is given in figure 6.1b. A shape deformation δS at coordinate $\mathbf{x}$ on the airfoil is applied in the direction normal to the surface $\mathbf{n}_S(\mathbf{x})$.

[1] This chapter is based on the journal paper: **J.H.S. de Baar, T.P. Scholcz, R.P. Dwight and H. Bijl**, *Exploiting Adjoint Derivatives in High-Dimensional Metamodels*, AIAA Journal **53**, 5 (2015).

6.1. Shape parameterization

Small variations of the geometry may result in significant changes of the airfoil drag. The smooth fourier-like shape functions $f_i(\cdot)$ for $i \in \{1, \ldots, \frac{d}{2}\}$ are defined by

$$\begin{aligned} \bar{x} &= \frac{x - c_0}{1 - 2c_0}, \quad x \in [c_0, 1 - c_0] \\ f_i(x) &= \sin(\pi \bar{x}) \frac{\sin(i\pi \bar{x})}{i}, \end{aligned} \tag{6.1}$$

where $c_0 = 0.15$ is chosen to exclude the leading and trailing edges from deformation. The surface normals are approximated using $\mathbf{n}_S = [0\ 1]^T$ and $\mathbf{n}_S = [0\ -1]^T$ on the suction- and pressure-side respectively. A plot of the shape functions $f_i(\cdot)$ for $d = 32$ is given in Figure 6.2. The shape deformations on the suction- and pressure-side of the airfoil are respectively:

$$\begin{aligned} \Delta y^{\text{suction}}(x) &= \sum_{i=1}^{\frac{d}{2}} \xi_i^{\text{suction}} f_i(x), \\ \Delta y^{\text{pressure}}(x) &= -\sum_{i=1}^{\frac{d}{2}} \xi_i^{\text{pressure}} f_i(x), \end{aligned} \tag{6.2}$$

with d the total number of shape parameters, and ξ_i are independent identically-distributed truncated Gaussian random variables with $\mu_\xi = 0$ and $\sigma_\xi = 0.005$ truncated to the interval $\xi \in [-0.0125,\ 0.0125]$.

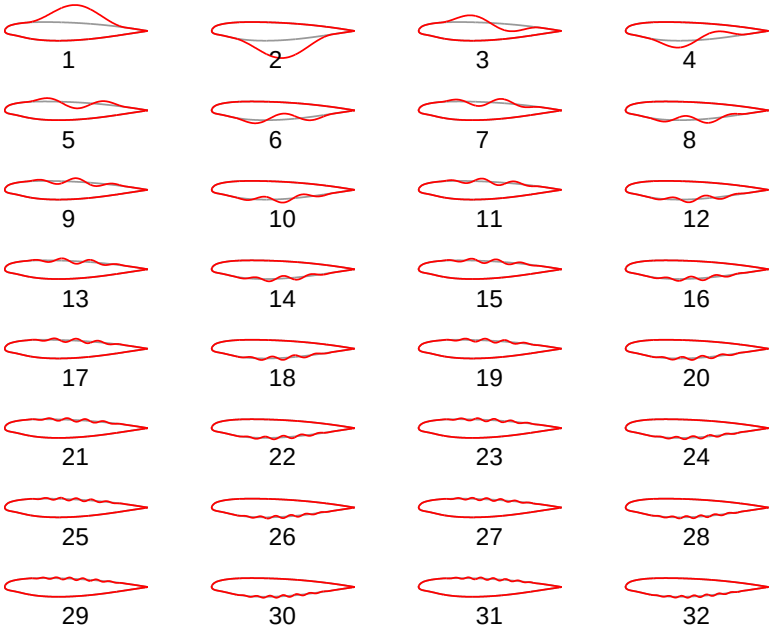


Figure 6.2: Shape functions for $d = 32$

6.2. Primary problem

This section contains a detailed description of the primary flow problem. To obtain sensitivities, adjoint equations are formulated from the equations describing the primary equations in Section 6.3. The solution of the primary problem together the adjoint problem enables the determination of the gradients which are subsequently used to obtain the Gradient-Enhanced Kriging meta-model.

6.2.1. Theory

Stanford's CFD solver SU^2 [2] is used to calculate the aerodynamic drag. The flow is described by the two-dimensional Euler equations of gas-dynamics with solid wall boundary conditions on the airfoil surface and appropriate characteristic-based far-field boundary conditions on the outer edges of the domain:

$$\begin{cases} \mathbf{R}(\mathbf{U}) = \nabla \cdot \mathbf{F} = 0 & \text{in } \Omega, \\ \mathbf{v} \cdot \mathbf{n}_S = 0 & \text{on } S, \\ \mathbf{U}_+ = \mathbf{U}_\infty & \text{on } \Gamma_\infty, \end{cases} \tag{6.3}$$

where the state variables are assembled in the vector $\mathbf{U}$ and $\mathbf{v}$:

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix} \quad \text{and} \quad \mathbf{v} = \begin{bmatrix} u \\ v \end{bmatrix}. \tag{6.4}$$

The flux vector $\mathbf{F}$ in Eq. (6.3) is given by

$$\mathbf{F} = \begin{bmatrix} \rho\mathbf{v} \\ \rho\mathbf{v} \otimes \mathbf{v} + \mathbf{I}p \\ \rho H\mathbf{v} \end{bmatrix}, \tag{6.5}$$

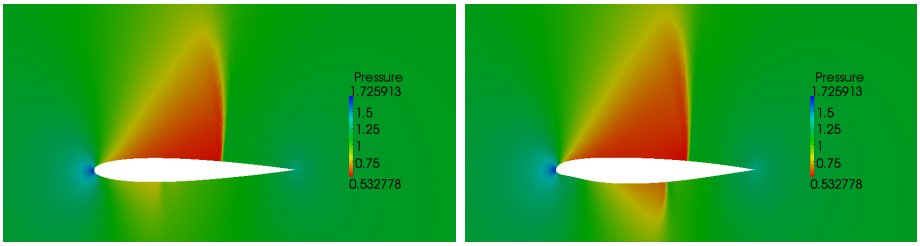
where the stagnation enthalpy is given by $H = E + \frac{p}{\rho}$ and the symbol $\otimes$ denotes the tensor product. The perfect gas assumption closes the system of equations in Eq. (6.3) using

$$p = (\gamma - 1)\rho \left[E - \frac{1}{2}(\mathbf{v} \cdot \mathbf{v}) \right]. \tag{6.6}$$

The Euler equations are discretized with an finite volume method on an unstructured grid. SU^2 is an edge-based solver and the JST scheme is selected for stabilization, see [2]. Artificial dissipation is computed using the differences in the undivided Laplacians and the conserved variables between the nodes on either end of the current edge, see [3, 4]. A pressure switch is used for triggering lower-order dissipation in the vicinity of shock waves. The airfoil drag is calculated using the surface integral

$$C_\mathrm{d} = \int_S \mathbf{d} \cdot (p\mathbf{n}_\mathbf{S}) ds \quad \text{with} \quad \mathbf{d} = \left(\frac{1}{\frac{1}{2}\rho_\infty V^2 c} \right) (\cos(\alpha), \sin(\alpha)) \,. \tag{6.7}$$

For the deformed airfoils, the grid deformation follows a spring analogy [3]. The baseline geometry of the FFAST airfoil has a drag coefficient of $C_\mathrm{d} \approx 0.0263$ at $\alpha = 1.25$ degrees and Mach = 0.8, the corresponding pressure field is shown in Figure 6.3b. The FFAST results are compared to the results obtained using a NACA0012 airfoil geometry. The baseline geometry of the NACA0012 airfoil has a drag coefficient of $C_\mathrm{d} \approx 0.0226$ for the same flow conditions, the corresponding pressure field is shown in Figure 6.3a.



(a) NACA0012 airfoil (b) FFAST airfoil

Figure 6.3: Baseline geometry pressure fields at $\alpha = 1.25$ degrees and Mach = 0.8.

In transonic airfoil design, the surface thickness of the pressure-side near the leading edge is often reduced in order to provide lift and positive pitching moment. This comes at the expense of the presence of a weak shock on the pressure-side,

as can be seen on the FFAST airfoil in Figure 6.3b. This "pressure-side shock" is not present on the symmetric NACA0012 airfoil, as can be seen in Figure 6.3a. Small variations of the airfoil geometry can have a large influence on the shock strength and therefore on the airfoil drag. The *continuous adjoint method* is used to quantify the sensitivities of the drag with respect to small variations of the geometry. The continuous adjoint method is chosen since it is available in the SU^2 code.

6.3. Continuous adjoint problem

The gradients of the drag with respect to the shape function parameters can be computed using the continuous adjoint method. The continuous adjoint method requires careful derivation of the adjoint equations in continuous form. The system is subsequently discretized and solved for the adjoint variables. The gradients are then obtained from surface integrals of adjoint variables on the airfoil surface. This section gives the theoretical development in Section 6.3.1 and some points of attention with the practical use of the method in Section 6.3.2.

6.3.1. Theory

The flow variables are constrained to satisfy the residual equations $\mathbf{R}(\mathbf{U}) = 0$. The airfoil drag can therefore be written as

$$C_\mathrm{d} = \int_S \mathbf{d} \cdot (p\mathbf{n_S}) ds + \int_\Omega \mathbf{\Psi}^T \mathbf{R}(\mathbf{U}) d\Omega, \tag{6.8}$$

where the *adjoint variables* are assembled in the vector $\mathbf{\Psi}$ and $\boldsymbol{\phi}$:

$$\mathbf{\Psi} = \begin{bmatrix} \psi_\rho \\ \psi_{\rho u} \\ \psi_{\rho v} \\ \psi_{\rho E} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\phi} = \begin{bmatrix} \psi_{\rho u} \\ \psi_{\rho v} \end{bmatrix}. \tag{6.9}$$

The first variation in airfoil drag δC_d due to a surface deformation δS is given by

$$\delta C_\mathrm{d} = \int_S (\mathbf{d} \cdot \nabla p) \delta S ds + \int_S (\mathbf{d} \cdot \mathbf{n}_S) \delta p ds + \int_\Omega \mathbf{\Psi}^T \delta \mathbf{R}(\mathbf{U}) d\Omega, \tag{6.10}$$

which takes the variation in flow equations ($\delta\mathbf{R}$, δp) into account, see [3]. With $\delta\mathbf{R} = \nabla \cdot \delta\mathbf{F}$ and $\delta\mathbf{F} = \frac{\delta\mathbf{F}}{\delta\mathbf{U}}\delta\mathbf{U}$ this becomes

$$\delta C_\mathrm{d} = \int_S (\mathbf{d} \cdot \nabla p) \delta S ds + \int_S (\mathbf{d} \cdot \mathbf{n}_S) \delta p ds + \int_\Omega \mathbf{\Psi}^T \nabla \cdot \mathbf{A} \delta\mathbf{U} d\Omega, \tag{6.11}$$

where the Jacobian of the flux vector is defined by $\mathbf{A} = \frac{\delta\mathbf{F}}{\delta\mathbf{U}}$. Using the product rule, Eq. (6.11) is expanded

$$\delta C_\mathrm{d} = \int_S (\mathbf{d} \cdot \nabla p) \delta S ds + \int_S (\mathbf{d} \cdot \mathbf{n}_S) \delta p ds + \int_\Omega \nabla \cdot \left[\mathbf{\Psi}^T \mathbf{A} \delta\mathbf{U}\right] d\Omega - \int_\Omega \nabla \mathbf{\Psi}^T \cdot \mathbf{A} \delta\mathbf{U} d\Omega. \tag{6.12}$$

Applying the divergence theorem to the third term in Eq. (6.12) to obtain

$$\int_{\Omega} \nabla \cdot \left[\boldsymbol{\Psi}^T \mathbf{A} \delta \mathbf{U}\right] d\Omega = \int_S \boldsymbol{\Psi}^T \mathbf{A} \cdot \mathbf{n}_S \delta \mathbf{U} ds + \int_{\Gamma_\infty} \boldsymbol{\Psi}^T \mathbf{A} \cdot \mathbf{n}_{\Gamma_\infty} \delta \mathbf{U} ds \tag{6.13}$$

$$= \int_S (\delta \mathbf{v} \cdot \mathbf{n}_S)(\rho \psi_\rho + \rho \mathbf{v} \cdot \boldsymbol{\phi} + \rho H \psi_{\rho E}) ds + \int_S [\mathbf{n}_S \cdot \boldsymbol{\phi} + \psi_{\rho E}(\mathbf{v} \cdot \mathbf{n}_S)] \delta p ds, \tag{6.14}$$

which is only true if the far-field boundary conditions are chosen in such a way that the second term in Eq. (6.13) vanishes.

$$\delta C_{\mathrm{d}} = \int_S \left[\mathbf{d} \cdot \mathbf{n}_S - \mathbf{n}_S \cdot \boldsymbol{\phi} - \psi_{\rho E}(\mathbf{v} \cdot \mathbf{n}_S)\right] \delta p ds + \int_S \left[\mathbf{d} \cdot \nabla p + \partial_n \mathbf{v} \cdot \mathbf{n}_S \vartheta + \nabla_S \cdot \mathbf{v} \vartheta\right] \delta S ds, \tag{6.15}$$

where $\vartheta = \rho \psi_\rho + \rho \mathbf{v} \cdot \boldsymbol{\phi} + \rho H \psi_{\rho E}$. The *adjoint* equations are given by

$$\begin{cases} \mathbf{A}^T \cdot \nabla \boldsymbol{\Psi} = \mathbf{0} & \text{in } \Omega \\ \mathbf{n}_S \cdot \boldsymbol{\phi} = \mathbf{d} \cdot \mathbf{n}_S - \psi_{\rho E}(\mathbf{v} \cdot \mathbf{n}_S) & \text{on } S \\ \mathbf{n}_{\Gamma_\infty}^T \cdot \mathbf{A}^T \boldsymbol{\Psi} = \mathbf{0} & \text{on } \Gamma_\infty. \end{cases} \tag{6.16}$$

Note that the partial differential equation in Eq. (6.16) is chosen such that the last volume integral in Eq. (6.12) is equal to zero. The solid wall and far-field boundary conditions in Eq. (6.16) are chosen such that the first term in Eq.(6.15) and the second term in Eq. (6.13) vanishes.

When the adjoint equations are satisfied, the surface sensitivity is given by

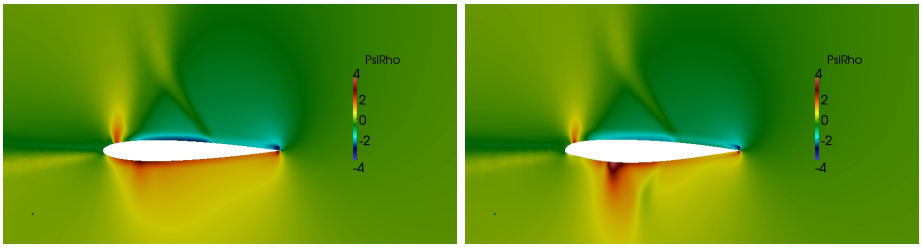
$$\delta C_{\mathrm{d}} = \int_S [\mathbf{d} \cdot \nabla p + \partial_n \mathbf{v} \cdot \mathbf{n}_S \vartheta + \nabla_S \cdot \mathbf{v} \vartheta] \delta S ds \tag{6.17}$$

$$= \int_S [\mathbf{d} \cdot \nabla p + (\nabla \cdot \mathbf{v}) \vartheta + \mathbf{v} \cdot \nabla \vartheta] \delta S ds. \tag{6.18}$$

The sensitivities with respect to the random variables ξ_i are found by substituting $\delta S = \xi_i \delta f_i + \delta \xi_i f_i$ with $\delta f_i = 0$ and $ds = dx$ in Eq. (6.18) to obtain

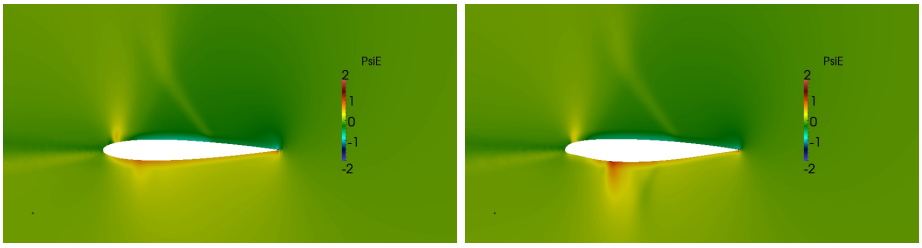
$$\frac{\partial C_{\mathrm{d}}}{\partial \xi_i} = \int_S [\mathbf{d} \cdot \nabla p + (\nabla \cdot \mathbf{v}) \vartheta + \mathbf{v} \cdot \nabla \vartheta] f_i dx \quad \text{for} \quad i \in \left[1, \ldots, \frac{d}{2}\right], \tag{6.19}$$

which should be evaluated for the shape functions at the suction and pressure side of the airfoil. The partial derivative notation is valid in the limit for small variations $\delta \to 0$. From Eq. (6.19) it becomes clear that only a single adjoint solve is required to obtain all random variable derivatives of interest.



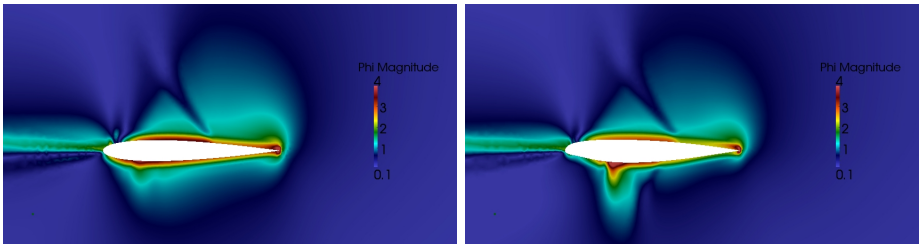
(a) NACA0012 airfoil (b) FFAST airfoil

Figure 6.4: Adjoint field ψ_ρ at α = 1.25 degrees and Mach = 0.8.



(a) NACA0012 airfoil (b) FFAST airfoil

Figure 6.5: Adjoint field $\psi_{\rho E}$ at α = 1.25 degrees and Mach = 0.8.



(a) NACA0012 airfoil (b) FFAST airfoil

Figure 6.6: Adjoint field $||\boldsymbol{\phi}||$ at α = 1.25 degrees and Mach = 0.8.

The adjoint fields ψ_ρ, $\psi_{\rho E}$ and the L^2-norm of $\boldsymbol{\phi}$ of the NACA0012 and FFAST airfoil are given in Figure 6.4, 6.5 and 6.6 respectively. Assessing the continuous adjoint solution offers physical insight into the problem. For example, a bump is present in the adjoint fields ψ_ρ, $\psi_{\rho E}$ and $||\boldsymbol{\phi}||$ at the location of the shock on the pressure-side of the FFAST airfoil. This bump contributes to the sensitivity integral in Eq. (6.19) via the adjoint variable $\vartheta = \rho\psi_\rho + \rho\mathbf{v}\cdot\boldsymbol{\phi} + \rho H\psi_{\rho E}$. From a physical point of view it is known that an increase of the airfoil thickness at this location results in an increase of the shock strength on the pressure-side. The drag of the

FFAST airfoil is therefore relatively sensitive to small variations of the pressure-side geometry near the leading edge.

6.3.2. Points of attention

The adjoint equations may suffer from stability issues due to local instability near sonic or stagnation points. A modified JST-type scheme is used on the adjoint variables in order to restore stability. The limiter is constructed by applying the Venkatakrishnan limiter to the adjoint density variable. In regions of high variability of the adjoint density variable additional artificial diffusion is added in order to stabilize the scheme.

The continuous adjoint approach assumes a continuously differentiable design surface. Sharp corners or edges which are inherently present in realistic geometries do not belong to this category. The local surface normal is not well defined in the vicinity of corners and edges leading to wrong estimates of the gradient. The solution is to exclude sharp corners and edges from the design surface such that they remain fixed in space. For example, by choosing $c_0 = 0.15$ the leading edge and sharp trailing edge of the FFAST airfoil are excluded from the design surface.

Two major points of attention of the continuous adjoint method that are often mentioned in literature (see, [5]) are

1. Differentiation of equations followed by discretization leads to an inconsistency between the computed gradient and the discrete implementation.
2. Boundary conditions for the adjoint variables are sometimes difficult to define due to the lack of physical interpretation of the adjoint variables.

The first problem should not be a problem when consistent discretizations are used such that the primary and adjoint solution converge. Verification of the code using a grid convergence study and finite difference checks of the adjoint gradients are therefore critical for the success of the continuous adjoint method. The second problem can be avoided by careful derivation of boundary conditions that naturally arise from the derivation of the adjoint system. The straightforward mathematical derivation of adjoint boundary conditions makes a physical interpretation obsolete.

6.4. Verification and error estimation

Both the primary implementation and the adjoint implementation need to be verified before reliable results are obtained. The primary implementation and adjoint implementation are verified in Section 6.4.1 and Section 6.4.2 respectively. In addition, noise that results from varying input parameters is estimated in Section 6.4.3.

6.4.1. Primary problem

In this section, a grid convergence study is presented for the matrix of test cases (NACA0012 & FFAST airfoil) $\times$ ($\alpha = 0$ & $\alpha = 1.25$). The only difference between the NACA0012 and the FFAST test case is the airfoil geometry.

A fine, medium and coarse unstructured grid are generated using the grid generator Gridgen, see table 6.1. The anisotropic meshing algorithm based on a surface

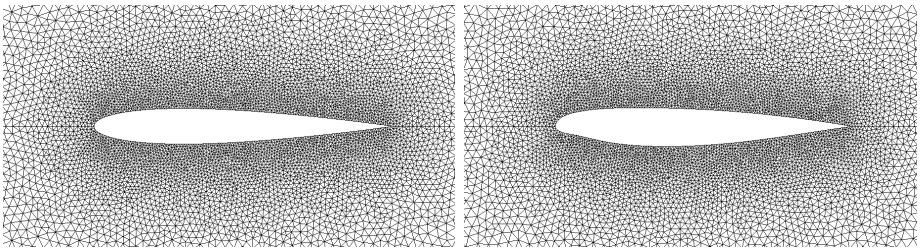
Mesh level	Name	Type	Number of vertices ($N_v = N^2$)
0	"Continuum"	unstructured	200.000
1	Fine	unstructured	50000
2	Medium	unstructured	20000
3	Coarse	unstructured	5000

Table 6.1: Overview of the grids used in the mesh refinement study.

deformation technique is used to construct the mesh of the lower- and upper-half of the flow domain, see [6]. The meshes of the NACA0012 airfoil case are symmetric about the horizontal axis. In this way, a non-lifting solution at zero degrees angle-of-attack is permitted, see Figure 6.7a . Define the effective grid refinement ratio as

$$r_{ij} = \sqrt{\frac{Nv_i}{Nv_j}}, \tag{6.20}$$

where Nv_i and Nv_j are the number of vertices of mesh level i and j respectively. The root in Eq. (6.20) is applied because the problem is two-dimensional.



(a) Medium grid, NACA0012 airfoil (b) Medium grid, FFAST airfoil

Let f_i and f_j denote the the Quantity of Interest computed at mesh level i and j respectively. The difference function is then defined by

$$\epsilon_{ij} = f_j - f_i. \tag{6.21}$$

The order of convergence p can then be computed by solving

$$\frac{\epsilon_{23}}{r_{23}^p - 1} = r_{12}^p \left[\frac{\epsilon_{12}}{r_{12}^p - 1} \right], \tag{6.22}$$

for p, see [7]. The Richardson extrapolated solution value is then obtained from

$$f^* = f_1 - \frac{\epsilon_{12}}{r_{12}^p - 1}. \tag{6.23}$$

The Grid Convergence Index is given by

$$\text{GCI}_{ij} = F_s \left(\left| \frac{\epsilon_{ij}}{f_i} \right| / (r_{ij}^p - 1) \right) \times 100, \tag{6.24}$$

where the factor of safety is taken as $F_s = 3$ which is a recommended value in [7]. The safety factor takes into account the uncertainty associated with non-asymptotic solutions.

$M = 0.8$	$\alpha = 0.0$		$\alpha = 1.25$	
	NACA0012	FFAST	NACA0012	FFAST
Coarse mesh	0.0165472328	0.0245940083	0.0304232966	0.0377445552
Medium mesh	0.0099000007	0.0174315033	0.0241053936	0.0286928993
Fine mesh	0.0086249986	0.0161120602	0.0228078869	0.0259552625
Extrapolated: f^*	0.0078172817	0.0153155676	0.0219139186	0.0226737524
Continuum	0.0083415160	0.0155171944	0.0225786653	0.0262502614
Observed: p	2.07	2.13	1.96	1.32
Medium GCI_{23}	63.11	36.42	27.27	62.93
Fine GCI_{12}	28.09	14.83	11.76	37.93

Table 6.2: Grid convergence results for (NACA0012 & FFAST airfoil) × ($\alpha = 0$ & $\alpha = 1.25$).

The grid convergence results of the test cases are assembled in table 6.2 . Asymptotic convergence with order p is observed from the results of three mesh levels with grid size $N = \sqrt{N_v}$, see Figure 6.7. It is however still uncertain whether the solutions are actually in the asymptotic regime. Continuum estimates of the drag are therefore used in order to assess asymptotic convergence. The continuum estimates of the NACA0012 airfoil are obtained from [8]. The continuum drag is calculated by taking the mean of the results obtained from the CFD packages FLO82, OVERFLOW v2.1t, CFL3Dv6, CFL3Dv6 + Vortex on a grid of $16.000.000$ vertices. The mean continuum drag of the NACA0012 airfoil is $C_\mathrm{d} = 0.0083415160$ for $\alpha = 0$ and $C_\mathrm{d} = 0.0225786653$ for $\alpha = 1.25$. The corresponding standard deviations are $\sigma = 4.197 \cdot 10^{-7}$ for $\alpha = 0$ and $\sigma = 1.186 \cdot 10^{-4}$ for $\alpha = 1.25$. The continuum drag of the FFAST airfoil is calculated using the continuum mesh of 200.000 vertices as defined in table 6.1.

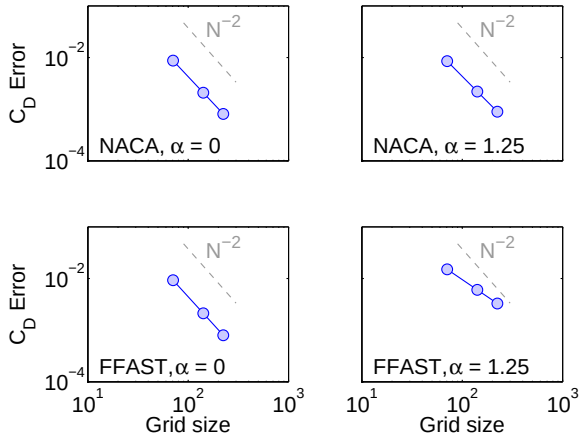


Figure 6.7: Observed order of convergence for (NACA0012 & FFAST airfoil) × ($\alpha = 0$ & $\alpha = 1.25$).

Asymptotic convergence is not observed with respect to the continuum estimates of the drag in table 6.2. Solving for p with too much uncertainty in the data can lead to wrong order estimates. Another negative effect on the order prediction is the error in the discrete function evaluations. Small errors in the drag due to the approximate integration of the pressure along the airfoil surface can significantly change the observed order of accuracy, see [8]. The Richardson extrapolates in table 6.2 under-predict the continuum estimates of the drag. This indicates that the true order of convergence is smaller than the observed order of convergence. CFD studies in [8] demonstrate that for typical CFD solvers actual asymptotic convergence is achieved starting from a mesh of 1.000.000 vertices. The purpose of the grid convergence study in this section is verification and error estimation and not accurate order estimation of CFD solvers. A detailed study to the order of convergence of SU^2 is therefore out of scope. The results of table 6.2 demonstrate convergence and are consistent with the NACA0012 continuum values reported in the literature. After successful verification, the FFAST test case with $\alpha = 1.25$ on the medium mesh is chosen for further computations. This test case is of interest since for $\alpha = 1.25$ the strong shock on the suction side of the airfoil and the relatively weak shock on the pressure side results in strong nonlinear behavior of the airfoil drag with respect to the shape parameters. It is exactly this behavior that is interesting from an uncertainty quantification point of view.

6.4.2. Adjoint problem

For the baseline geometry, the adjoint derivatives are shown in Figure 6.8, for the coarse, medium and fine grid. The gradients do not show clear monotone grid convergence, and therefore application of Richardson extrapolation is inappropriate. Lack of neat grid-convergence behavior is typical for derivatives, and can sometimes be traced back to specific causes, e.g. shocks moving in relation to grid-lines [9]. Pragmatically we must accept this situation.

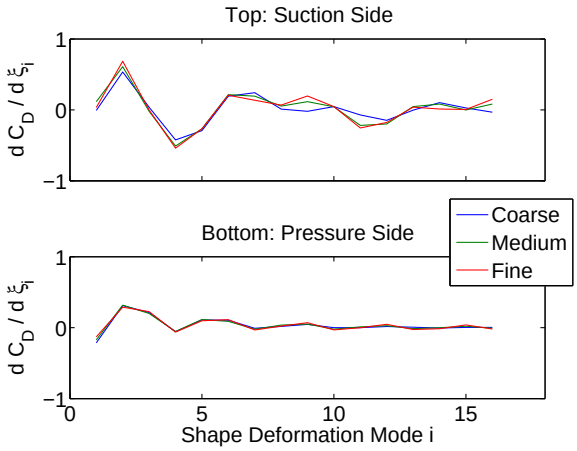


Figure 6.8: Grid convergence of the gradients.

The adjoint-based gradients are verified using a finite difference check. The finite difference gradients are computed with a stepsize $h = 10^{-5}$, resulting in a truncation error of $\approx 10^{-4}$. Two random shape parameters are used on a set of 40 random samples. The gradient error is defined as

$$e = \nabla_{\text{Adjoint}} - \nabla_{\text{FD}}. \tag{6.25}$$

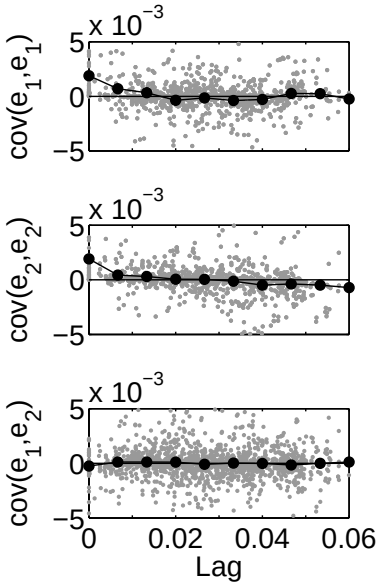


Figure 6.9: Spatial dependence of the covariance of the gradient error, for two random shape parameters. Observed covariance has been binned for different lags.

In Figure 6.9, we verify that the gradient error is spatially uncorrelated. We have binned the sample pairs by lag (*i.e.* sample separation distance), and computed the covariance. The self-covariance of both e_1 and e_2 is roughly zero for lag > 0 (the covariance corresponds to ϵ_∇^2 for lag $= 0$), while the cross-covariance is roughly zero for all lags. Therefore, the assumption that the gradient errors are spatially uncorrelated is verified for this case.

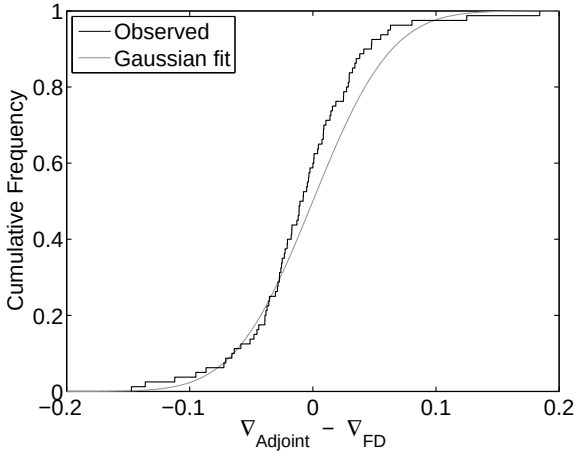


Figure 6.10: Cumulative histogram of the gradient error.

In Figure 6.10, we verify that the gradient error is normally distributed, with a standard deviation of ≈ 0.05. Note that this standard deviation is significantly larger than the truncation error of the finite difference gradients.

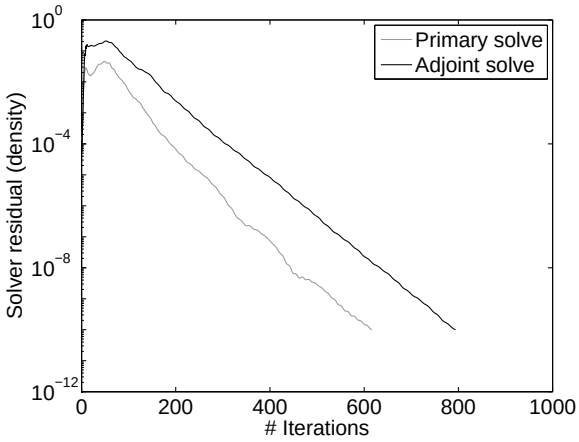


Figure 6.11: Iterative convergence of the residual, associated with the density.

Figure 6.11 illustrates the iterative convergence of the solver for the baseline design. The tolerance of the residual associated with the density has been set to 10^{-10}. After an initial bump, both the primary and adjoint solve show clear exponential convergence up to residual tolerance. Although ideally the primary and adjoint solve should show the same convergence rate, we consider the rates sufficiently close for code verification purposes.

Summarizing, we verify that the solver residual converges exponentially, there is sufficient grid convergence, and the gradient errors are uncorrelated and normally distributed.

6.4.3. Noise estimation

Figure 6.12 shows CFD results when varying a single shape parameter, for a deformation on the pressure side.

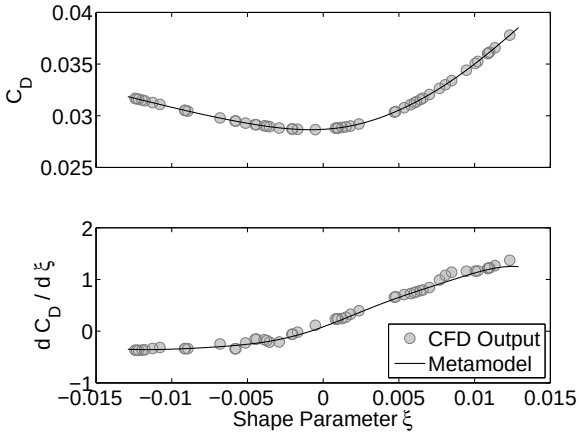


Figure 6.12: Drag coefficient and gradient for a single random shape parameter: note the high level of noise in the gradients.

We observe minimum of the drag coefficient C_d close to $\xi = 0$, which corresponds to the baseline geometry; this is because the airfoil has been optimized for these conditions. Note that the noise in the derivative $\partial C_\mathrm{d}/\partial\xi$ is larger than in C_d. The noise in both can be estimated within the Kriging framework assuming a certain smoothness of the true response and unbiased Gaussian noise. In particular we maximize the log likelihood [10, Eq. 29]:

$$\ln p(\epsilon, \epsilon_\nabla | \mu, \sigma, \mathbf{y}) = -\ln|A| - (\mathbf{y} - \boldsymbol{\mu})^T A^{-1} (\mathbf{y} - \boldsymbol{\mu}), \tag{6.26}$$

with gain matrix $A = R + HPH'$. This maximum likelihood estimate (MLE), illustrated in Figure 6.13, results in an estimated standard deviation of $\epsilon \approx 6 \times 10^{-6}$ and $\epsilon_\nabla \approx 5 \times 10^{-2}$ in the noise of the drag and drag-derivative respectively.

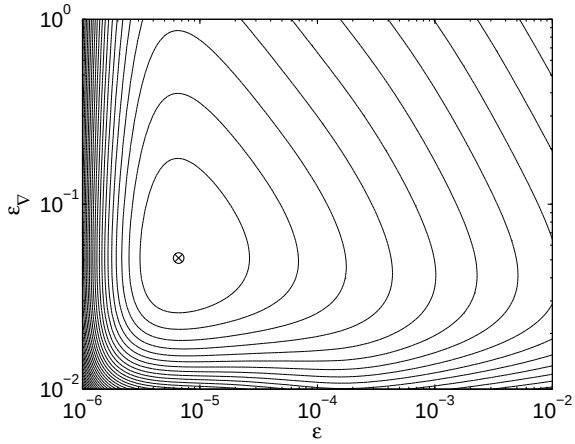


Figure 6.13: Joint likelihood of the standard deviation of the noise in the drag coefficient and adjoint-based gradients.

Using a GEK metamodel, estimated noise level in derivatives can be specified via the matrix R in (4.2–4.3). The quality of the metamodel depends on this parameter corresponding to the true level of derivative noise. Figure 6.14 shows the RMS error of the metamodel for four random shape parameters and 16 samples, as a function of the specified derivative noise. For zero and very low values the quality of the metamodel is clearly degraded as noisy data is overfitted. At the other extreme, when very large data noise is assumed, the GEK predictor effectively ignores the derivative information and approaches the basic Kriging metamodel. In between there is a limited range for which an improved metamodel is obtained, the best metamodel being close to the maximum likelihood estimate of $\epsilon_\nabla \approx 5 \times 10^{-2}$.

This last four-dimensional result compares well with the standard deviations found from the finite difference verification in two dimensions and from the MLE in 1 dimension, which is an indication that this value is consistent and can be used for higher dimensions, as we will assume in the following.

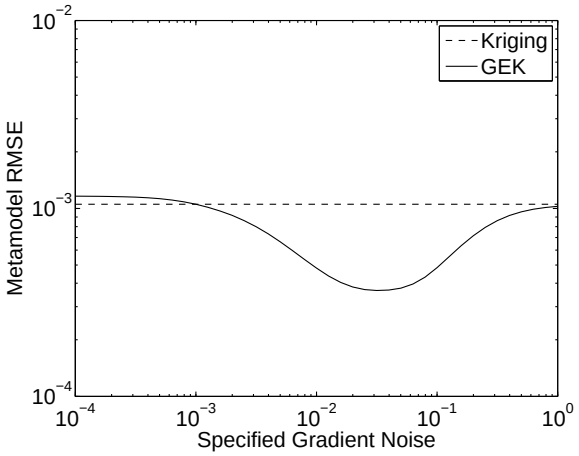


Figure 6.14: RMS error of Kriging and GEK metamodels as a function of specified derivative noise for $d = 4$.

6.5. Numerical experiments

We evaluate the quality of the metamodel, which can be quantified by the root mean squared prediction error:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N_{\mathrm{MC}}} \sum_i \left\{ \mathrm{E}(C_{\mathrm{D},i} | \mathbf{C}_\mathrm{D}, \nabla \mathbf{C}_\mathrm{D}) - C_{\mathrm{D,ref},i} \right\}^2}, \tag{6.27}$$

which is the Monte Carlo approximation [11] of the prediction error:

$$\sqrt{\int_\Omega p(\xi) \left\{ \mathrm{E}(C_\mathrm{D} | \mathbf{C}_\mathrm{D}, \nabla \mathbf{C}_\mathrm{D}) - C_{\mathrm{D,ref}} \right\}^2 \mathrm{d}\xi}, \tag{6.28}$$

where $p(\xi)$ is the probability distribution of the random shape parameters, $\mathrm{E}(C_\mathrm{D}|\mathbf{C}_\mathrm{D}, \nabla\mathbf{C}_\mathrm{D})$ is the metamodel prediction conditional on the discrete solver output $[\mathbf{C}_\mathrm{D}, \nabla\mathbf{C}_\mathrm{D}]$, and $C_{\mathrm{D,ref}}$ are $N_\mathrm{MC} = 1,000$ reference samples. For most dimensions, $1,000$ reference samples were sufficient to obtain a reliable RMS error. The number of samples for the design of experiment is typically much lower than the number of reference samples. The number of Latin- Hypercube Samples for the design of experiment was taken to be $N_\mathrm{LHS} = 200 - 400$. An overview of the total numerical experiment is given in Table 6.3.

d	sampling method	number of samples	gradients
1	MC	1000	no
1	LHS	400	yes
2	MC	500	no
2	LHS	200	yes
3	MC	1000	no
3	LHS	400	yes
4	MC	500	no
4	LHS	200	yes
6	MC	500	no
6	LHS	200	yes
8	MC	500	no
8	LHS	200	yes
12	MC	1000	no
12	LHS	400	yes
16	MC	1000	no
16	LHS	400	yes
24	MC	1000	no
24	LHS	400	yes
32	MC	1000	no
32	LHS	400	yes

Table 6.3: An overview of the numerical experiments for each dimension. Reference samples are obtained with Monte Carlo (MC) and Latin- Hypercube Sampling (LHS) is used for the metamodel design of experiment.

For these numerical experiments, the following quantities are of interest:

- Number of solves required to reach a specified target accuracy (solves to target)
- Speedup of the RMSE obtained with GEK with respect to Kriging (speedup)
- Speedup of statistics obtained with GEK with respect to Kriging (speedup for various statistics)

The results are discussed in section 6.5.1 and 6.5.2.

6.5.1. Solves to Target

Figure 6.15 shows the number of solves required to reach a given target RMS error of 0.004 in the metamodel, assuming an appropriate value for the derivative error. The observed CPU ratio for this test case is

$$\eta_{\text{CPU}} = \frac{t_{\text{val}} + t_{\text{grad}}}{t_{\text{val}}} \approx 2.4,$$

independent from the number of dimensions and sufficiently close to the ideal value $\eta_{\text{CPU}} = 2$. The speedup is therefore computed from the number of solves to target in order to test if the gradients are as informative as values for the metamodel. Latin-hypercube sampling (LHS) is used, and the (random) sampling is repeated 50 times to explore sensitivity to the sampling plan. The symbols show the median, with vertical bars indicating 20% to 80% quantiles.

The curse of dimensionality is observed: when we increase the number of random shape parameters, the number of CFD solves increases exponentially, for both Kriging and GEK. However as expected for higher dimensions, GEK appears increasingly efficient when compared to Kriging.

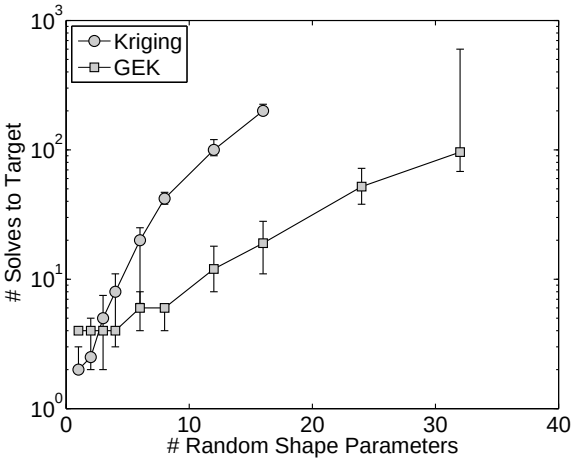


Figure 6.15: Number of solves required for a given target RMS error of the metamodel. Median and 20% – 80% quantiles with respect to sampling plan.

Recasting these results in terms of speedup of GEK with respect to Kriging, now for a range of target accuracies:

$$[0.0050,\ 0.0047,\ 0.0044,\ 0.0041,\ 0.0038,\ 0.0035,\ 0.0032,\ 0.0029,$$
$$0.0026,\ 0.0023,\ 0.0020,\ 0.0017,\ 0.0014,\ 0.0011,\ 0.0008,\ 0.0005],$$

Figure 6.16 illustrates that the theoretical result in Eq. (4.1) (grey line) agrees with the numerical experiment to a remarkable accuracy—but only if a range of target accuracies, derivative noise, and sampling plan variability are accounted for.

This supports the hypothesis that derivatives are as informative as values, and the expected benefits of using derivatives can be attained.

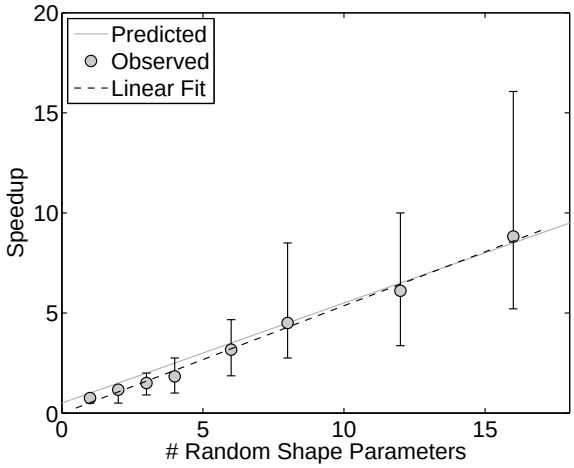


Figure 6.16: Factor speedup of GEK over Kriging. Solid line: prediction based on the theoretical result in Eq. (4.1), circles: numerical experiments based on transonic airfoil, dashed line: linear regression fit.

Figure 6.17 illustrates how the speedup depends on the specified gradient noise, for a single LHS-design with four random shape parameters. The optimum level of specified gradient noise depends on the particular LHS-design, and would ideally be found from an adjoint code that provides estimates of gradient error.

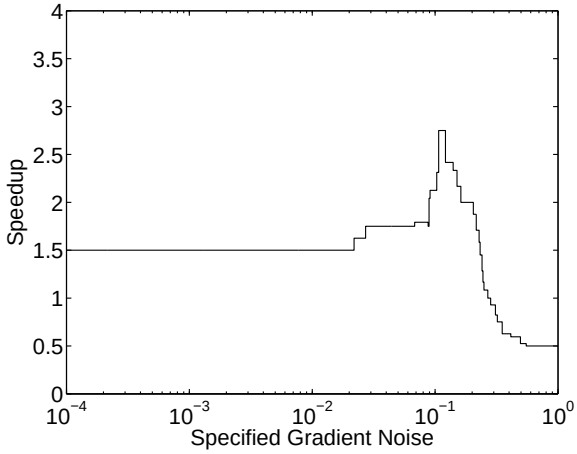


Figure 6.17: Speedup versus specified gradient noise, observed for an individual LHS-design of four random shape parameters. Median and $20\% - 80\%$ quantiles with respect to target accuracy and sampling plan.

6.5.2. Speedup for Various Statistics

In uncertainty propagation, we are often concerned with one or more statistics of the quantity of interest. Figure 6.18 shows the median speedups for the mean, variance, and skewness of the drag coefficient—again for a range of target accuracies. Also, it shows the median speedup for a probability of failure p_f: the probability that the drag coefficient exceeds a certain operational limit. The operational limit is chosen such that $p_f \approx 0.2$. The observed accuracies are again computed from $N_{\mathrm{MC}} = 1,000$ reference samples.

Since the speedups for these statistics depend on the form of the response as well as on the probability distributions of the random shape parameters, we can not provide a simple predicted speedup in the form of Eq. (4.1). Numerical experiments show that all statistics show a linear speedup—although the slopes differ.

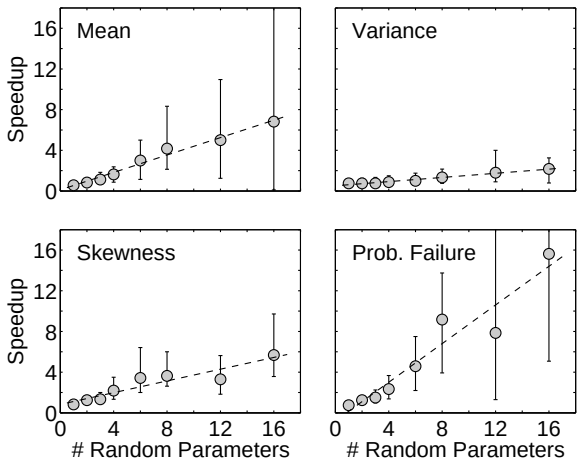


Figure 6.18: Speedups for statistics of the drag coefficient, medians with linear regression fit.

6.6. Conclusions

We observe the predicted reduction in the number of samples required to build a metamodel when using adjoint derivative information. In particular we observe in numerical experiments a speedup of GEK over Kriging of a factor of $S = \frac{1}{2}(1+d)$, for up to 16 shape parameters for a transonic airfoil. Additionally we observe linear speedups for the mean, variance, skewness, and probability of failure of the drag coefficient.

An important requirement for the succesful use of gradient information in general (and in GEK in particular) has been shown to be the correct specification of gradient noise. Even in this simple 2d inviscid case moderate gradient noise is observed, and failing to account for this causes the added gradient information to degrade the metamodel. While gradient noise can be estimated with a MLE procedure (at the cost of some information in the samples), and while GEK is somewhat robust to under- or overspecification of gradient noise, there is clearly a need for

adjoint codes to provide estimates of gradient error. Methods to achieve this do not exist presently.

References

[1] D. Jones and A. Gaitonde, *Future fast methods for loads calculations: The 'FFAST' project,* in *Innovation for sustainable aviation in a global environment*, edited by D. Knörzer and J. Szodruch (IOS Press BV, 2012) pp. 110 – 115.

[2] F. Palacios, J. Alonso, K. Duraisamy, M. Colonno, J. Hicken, A. Aranake, A. Campos, S. Copeland, T. Economon, A. Lonkar, T. Lukaczyk, and T. Taylor, *Stanford university unstructured (SU^2): An open-source integrated computational environment for multi-physics simulation and design,* AIAA Paper 2013-0287, 51th AIAA Aerospace Sciences Meeting and Exhibit. January 7th - 10th (Grapevine, Texas, USA, 2013).

[3] D. Economon, F. Palacios, and J. J. Alonso, *Optimal shape design for open rotor blades,* AIAA Paper 2012-3018, 30th AIAA Applied Aerodynamics Conference. June 25th - 28th .

[4] D. J. Mavriplis, *Accurate multigrid solution of the Euler equations on unstructured and adaptive meshes,* AIAA Journal **28**, 213.

[5] G. Carpentieri, *An Adjoint-Based Shape-Optimization Method for Aerodynamic Design*, Ph.D. thesis (2009).

[6] J. P. Steinbrenner and J. P. Abelanet, *Anisotropic Tetrahedral Meshing Based on Surface Deformation Techniques,* 45th AIAA Aerospace Sciences Meeting and Exhibit 8-11 January 2007, Reno, Nevada.

[7] P. J. Roache, *Verification and Validation in Computational Science and Engineering* (Hermosa Publishers, 1998).

[8] J. C. Vassberg and A. Jameson, *In pursuit of grid convergence for two-dimensional Euler solutions,* Journal of Aircraft **47**, 1152 (2010).

[9] M. B. Giles, M. C. Duta, J. D. Muller, and N. A. Pierce, *Algorithm developments for discrete adjoint methods,* AIAA Journal **41**, 198 (2003).

[10] J. H. S. de Baar, R. P. Dwight, and H. Bijl, *Improvements to Gradient-Enhanced Kriging using a Bayesian Interpretation,* International Journal for Uncertainty Quantification **4**, 205 (2014).

[11] N. Metropolis and S. Ulam, *The Monte Carlo Method,* J. Am. Stat. Assoc. , 335 (1949).

7

Conclusions and Recommendations

7.1. Conclusions

The cost and turnaround time of the load calculation cycle in the design process of aircraft can be reduced by developing new efficient numerical simulation techniques. This leads to a decrease in wind tunnel testing requirements, in-flight testing requirements and risk of design modifications, amongst others.

To this end, efficient prediction of steady and unsteady force coefficients in the flight envelope is of utmost importance. Especially load cases due to gust and manoeuvres are of interest since they determine the most extreme stress levels, fatigue damage and damage tolerance during the design cycle. This requires time-accurate analysis of fluid-structure interaction. Uncertainties in flight conditions, manoeuvres, shape and material properties require furthermore the use of methods that quantify this uncertainty. Efficient algorithms for high-fidelity simulation of fluid-structure interaction and uncertainty quantification are therefore desired in industry.

Multi-fidelity algorithms can be used in order to reduce the computational cost of fluid-structure interaction simulations and uncertainty quantification methods. Two important areas of research are:

- Multi-fidelity acceleration of high-fidelity models.
- Multi-fidelity analysis for uncertainty quantification.

This thesis aimed at developing efficient multi-fidelity algorithms for fluid-structure interaction and uncertainty quantification.

7.1.1. Multi-fidelity acceleration of high fidelity fluid-structure interaction

Space-mapping is a general multi-fidelity technique - originating from the field of multi-fidelity optimization - that can be applied to a wide variety of engineering problems. In this dissertation it is demonstrated that space-mapping can be applied in order to accelerate strongly coupled partitioned fluid-structure interaction. Aggressive Space-Mapping has been successfully applied to three academic fluid-structure interaction problems of increasing complexity in order to speed up the partitioned algorithm. The main findings are

- The estimated and observed speedup of the Aggressive Space-Mapping algorithm with respect to the Quasi-Newton algorithm is larger than 1 for most test cases considered.
- The influence of the time step on the achieved speedup is large compared to other parameters. In case of compressible flows the speedup can rise to 1.5 for large time-steps . For incompressible flows the fluid-structure interaction is much stronger and for this reason higher speedups can be obtained.

The space-mapping framework allows for the design of new coupling algorithms that - using the information of simplified models - can significantly speedup partitioned strongly coupled FSI simulations. Off the shelf solvers can be reused such that the method is suited for industry.

7.1.2. Multi-fidelity analysis for uncertainty quantification

Gradient-Enhanced Kriging (GEK) is a response surface technique that can be used for uncertainty quantification. In this thesis, GEK is applied to two academic test cases in order to assess the speedup with respect to ordinary Kriging as a function of the number of design variables d. When the gradients are obtained with the adjoint method and when it is assumed that the derivative is as informative as the quantity of interest, a theoretical speedup S can be derived as $S \approx \frac{1}{2}(1+d)$. The theoretical speedup is compared to the actual observed speedup for the panel divergence problem and the FFAST airfoil problem.

Panel divergence problem

The quantity of interest is the critical Mach number, for which panel divergence occurs. The observed speedup of GEK as compared to Kriging increases with the number dimensions in the problem. The observed speedup is however lower than the theoretical prediction. This can be explained by the fact that the quantity of interest depends only weakly nonlinear on the random parameters in the problem. In that case, the assumption that the derivatives are as informative as the quantity of interest does not hold.

FFAST airfoil problem

The quantity of interest is the airfoil drag for the FFAST airfoil at transonic conditions. The speedup of GEK as compared to Kriging agrees well with the theoretical

speedup $S = \frac{1}{2}(1 + d)$. However, this is only true if the variation of the sampling plan, the derivative noise and the target accuracies are accounted for. The speedups for the mean, variance, skewness and the probability of failure of the drag coefficient are observed to be linear. However, no theoretical results for the slope of these lines are found. An important requirement for the success of GEK is the correct specification of gradient noise. Failing to account for gradient noise can degrade the metamodel instead of improving it.

7.2. Recommendations

The following recommendations for future research are proposed:

7.2.1. Different space-mapping strategies

As was pointed out in the introduction, Aggressive Space Mapping is not the only space-mapping method that exists. Output Space-Mapping and Manifold Mapping are both promising multi-fidelity coupling strategies. Some research in this direction has been performed but it is still not clear which method results in the most efficient and robust coupling algorithm. The comparison of the performance might depend on the type of low-fidelity model used in the space-mapping algorithm and it is well possible that the choice of low-fidelity model dictates the best space-mapping strategy. This is an interesting topic for further research.

7.2.2. Different low-fidelity models in space-mapping

Since space-mapping is a black-box coupling technique any low-fidelity model can be used in the coupling algorithm. In this dissertation mainly simplified fluid operators are used or fluid models discretized on coarser grids. Reduced Order Models (ROMs) using a projection on a reduced basis are not considered in this work. It would be interesting to start with a reduced basis fluid-structure interaction ROM for the fluid in the space-mapping algorithm, thereby accelerating the coupling algorithm. After a number of converged high-fidelity fluid-structure interaction time steps the new high-fidelity time history could be used to train the Reduced Order Model, i.e. update the ROM basis for the next time interval. This process could continue until in the end both the high fidelity time-history and the trained ROM are available on the time interval of interest and it would be cheaper than first computing the high-fidelity fluid-structure interaction on the time interval and constructing the Reduced Order Models afterwards.

7.2.3. Performance of GEK on more complex test cases

For the test cases considered in this dissertation, Gradient Enhanced Kriging shows a speedup with respect to Kriging. In order for GEK to be successful, gradient noise should be taken into account. The gradient errors are assumed to be independent and Gaussian with zero mean and the standard deviation resembling the magnitude of the error. This is not true if the errors are correlated or if the errors contain a systematic component. This is likely the case when Navier-Stokes solvers are used which often adopt a frozen turbulence assumption in the adjoint formulation.

Demonstrating the speedup of GEK on a more complex test case, i.e. Navier-Stokes on a three-dimensional wing, would therefore be interesting. Furthermore, including the systematic error in the hyperparameter estimation could improve the performance of GEK in this case.

Acknowledgements

The research presented in this thesis is the result of 4 years work. Here, I would like to thank all people that helped and supported me during this journey.

I would like to thank my supervisors Hester Bijl and Alexander van Zuijlen for giving me the opportunity to do my PhD on an interesting topic and for all the progress meetings of the past 4 years. Next, I would like to thank the partners of the FFAST project for all the interesting presentations during the project meetings and for their helpful advice.

Many thanks go to the co-workers who contributed to this thesis. Firstly, I would like to thank Jouke de Baar for the inspiring and smooth collaboration. Thank you for the work sessions that ranged from Hotel New York to the SS Rotterdam. Not to forget the ”blikjes” and the dinners in the ”ballentent”. Secondly, I would like to thank Richard Dwight for his critical and motivating feedback. Thirdly, Clemens Verhoosel, thank you for the open academic discussions we had and your willingness to share your expertise in the field of uncertainty quantification. And finally, many thanks go to Carlos Baptista and Liesbeth Florentie who I supervised during their master thesis work.

Furthermore, I thank all my colleagues at the Aerodynamics department for the enjoyable time I had, especially Joost, Jasper, Peter, Vahid, Sander and Ferry.

I would like to acknowledge my parents, José van Bergen and Peter Scholcz and my siblings Lucas, Sophie and Olga. Thank you for being there for me.

Special thanks go to Ramon, Jornt, Maikel, Annemarie and Patricia for supporting me in difficult times. Without you I would have never been able to finish this work.

Thomas Scholcz
Delft, July 2015

Curriculum Vitæ

Thomas Peter Scholcz

30-03-1983	Born in Nieuwegein, the Netherlands
2002–2005	B.Sc, Aerospace Engineering Delft University of Technology
2005–2008	M.Sc, Aerospace Engineering Delft University of Technology *Track:* Aerodynamics *Thesis:* Stochastic analysis of fluid-structure interaction
2009–2010	Researcher Peutz B.V.
2010-2014	PhD Aerospace Engineering Delft University of Technology *Thesis:* Multi-fidelity methods for fluid-structure interaction and uncertainty quantification *Promotor:* Prof. dr. ir. drs. H. Bijl
2014-2015	Applied Computational Fluid Dynamics (CFD) specialist Maritime Research Institute Netherlands (MARIN)

List of Publications

Journal articles

- **C.V. Verhoosel, T.P. Scholcz and S.J. Hulshoff**, *Uncertainty and reliability analysis of fluid-structure stability boundaries*, AIAA Journal **47**, 91 (2009).
- **T.P. Scholcz, A.H. van Zuijlen and H. Bijl**, *Space-mapping in fluid-structure interaction problems*, Computer Methods in Applied Mechanics and Engineering **281** (2014).
- **J.H.S. de Baar, T.P. Scholcz, R.P. Dwight and H. Bijl** , *Exploiting adjoint derivatives in high dimensional metamodels*, AIAA Journal **53**, 5 (2015).

Conference contributions

- **C.V. Verhoosel, T.P. Scholcz, S.J. Hulshoff and M.A. Gutiérez** , *Transient stability analysis of an aeroelastic problem with random fluid and structure properties*, 10th International Conference on Structural Safety and Reliability (ICOSSAR 10), Osaka, Japan (2009).
- **T.P. Scholcz, A.H. van Zuijlen and H. Bijl**, *A multi-model incremental adaptive strategy to accelerate partitioned fluid-structure algorithms using space-mapping*, In M Papadrakakis, E Onate & B Schrefler (Eds.), Computational Methods for Coupled Problems in Science and Engineering IV (pp. 1-13). Barcelona, Spain: International Center for Numerical Methods in Engineering (CIMNE) (2011).
- **T.P. Scholcz, A.H. van Zuijlen and H. Bijl**, *Accelerated partitioned fluid-structure interaction using space-mapping.*, In Proceedings of the 10th World Congress on Computational Mechanics (WCCM 2012) (pp. 1-15). Sao Paolo, Brazil (2012).
- **J. H. S. de Baar, T. P. Scholcz, C. V. Verhoosel, R. P. Dwight, A. H. van Zuijlen and H. Bijl**, *Efficient uncertainty quantification with gradient-enhanced kriging: applications in FSI*, 6th European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS 2012, Vienna (2012).